Three Steps to Use the On-Line Help

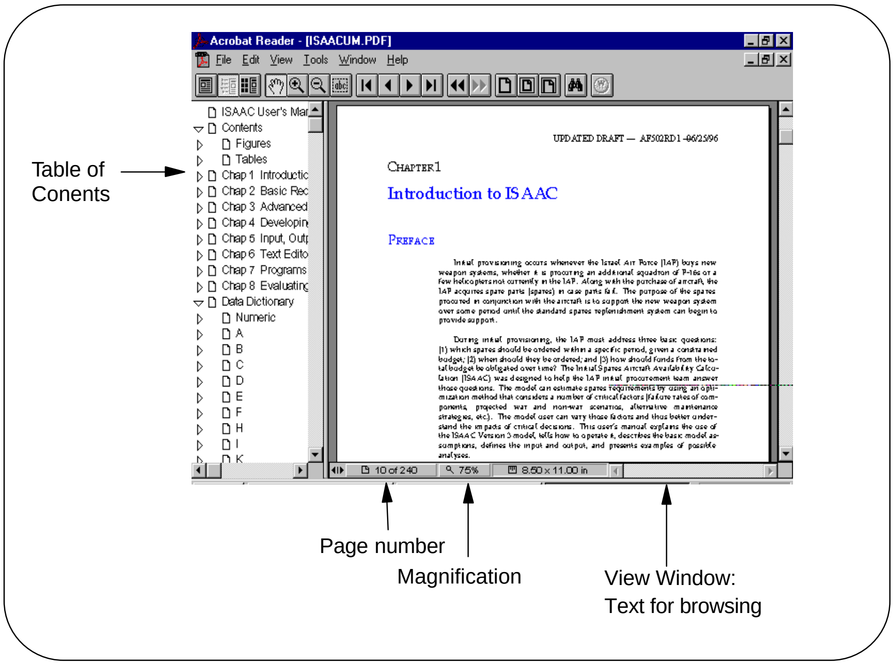
1. Find the section you want to view in the Table of Contents (bookmarks) on the left of this screen.

2. Expand the section by clicking on the ▷ symbol to its left — revealing its subsections.

3. Click on the specific sub-section you want. This will bring the text of that section immediately to your view (right side of the screen).

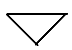
The following 4 pages summarize the most useful Help features such as printing, looking up words in the dictionary, using the icons displayed on the toolbar (above), and changing the display. We have summarized the key features, even though they are explained in the Help pull-down menu, because it is difficult to find specific information using that Help system.

1. Structure. This document should open with a Table of Contents (TOC) section to the left of this page. The TOC consists of bookmarks to bring you to each section of the document and is the recommended location for moving quickly to particular sections in the user's manual.

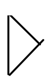


2. The triangles in the TOC section indicate whether the bookmark has any sub-bookmarks and if any of them are displayed. To display the next level of sub-bookmarks click on the triangle. No triangle indicates that the bookmark has no sub-bookmarks.

a. A triangle pointing down indicates that sub-bookmarks are open.



b. A triangle pointing to the right indicates that sub-bookmarks exist but they are not displayed.



3. Navigation. To move to a particular section in the user's manual click on the bookmark for that section. (Alternate methods include using the page down key or scroll to the section using the Windows vertical scroll bar.)

4. Printing.

a. Before printing select Printer Setup from the File pull-down menu and make sure the program is set to print to the printer you want.

b. To print one or more pages of the on-line user's manual, select Print from the File pull-down menu. This will open the print dialog box. You can print all pages, the current page, or number is displayed at the lower left of the view window.) Make your selections and choose print.

5. Organization. The on-line manual is essentially the same as the written version of the user's manual. However, the bookmarks have been set up to take advantage of the automated format. The bookmarks are organized into the following three groups:

a. Contents consists of the Table of Contents as well as Table and Figure bookmarks which consist of individual bookmarks that will bring you to any table or figure in the document.

b. Chap 1-8 consists of the eight chapters of the user's manual. The chapters are presented in sequence and can be viewed consecutively if desired.

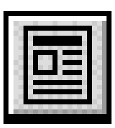
c. Data Dictionary consists of definitions for each of the field names, window names or screen objects of ISAAC. The dictionary has been organized alphabetically with all names stored under their respective first letters. The Number section contains those field names that begin with numerical values.

Help Tools and Buttons

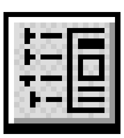
The toolbar contains tools for selecting, viewing, annotating, and linking documents. Select a tool by clicking the tool icon.

To hide or show the toolbar, choose Hide Toolbar or Show Toolbar from the Window menu. The command alternates between the two options.

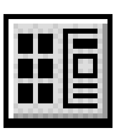
The toolbar contains the following tools and buttons



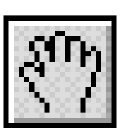
The **Page Only** button closes the overview area of the window.



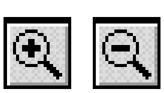
The **Bookmarks and Page** button opens the overview area and displays bookmarks created for the document. Click a book-mark's name to go to the location marked by that bookmark. See for more information.



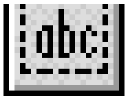
The **Thumbnails and Page** button opens the overview area and displays thumbnail images of each document page. Click a thumbnail to go to the page marked by that thumbnail. See for more information.



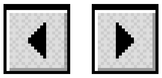
The **hand** tool enables you to move a single document page on the screen when the page does not fit within the main window. Drag the hand tool in the direction you want to move the page.



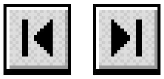
The **zoom** tools magnify and reduce the page display. See for more information.



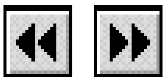
be copied to the Clipboard by using the Copy command. See



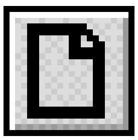
The **Previous Page** and **Next Page** buttons move the document backward or forward one page at a time. See



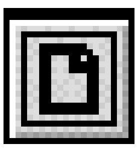
The **First Page** and **Last page** buttons move the document to the first or last page of a document. See



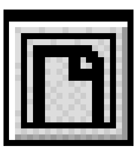
The **Go Back** and **Go Forward** buttons retrace your steps through a document, moving to each view in the order visited. Go Back also returns you to the original document when you click a link to another document. See
for more information.



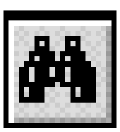
The **Actual Size** button displays the page at 100 % magnification.



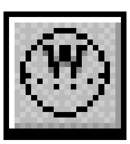
The **Fit Page** button scales the page to fit within the window.



The **Fit Width** button scales the page to fill the width of the window. Holding down Control (Windows) or Option (Macintosh) and clicking the Fit Width button scales the page so that the visible text and graphics fit the width of the window.



The **Find** button searches for part of a word, a complete word, or multiple words in a document. See r more information.



The **Web Browser** button starts your Web browser. If your Web browser is already running, it becomes the active application.

Initial Spares Aircraft Availability Calculation — ISAAC

Version 3.91

User's Manual

AF502RD1

July 1996

Robert C. Kline
Stephen P. Ford
Thomas Meeks
F. Michael Slay
Frank L. Eichorn

Contents

Chapter 1. Introduction to ISAAC 1-1

Preface 1-1

Overview of ISAAC 1-2

Factors Considered by the Weapon System Approach 1-2

Capabilities Provided by the ISAAC Interface 1-4

Installing ISAAC 1-4

Conventions 1-4

Loading ISAAC and Supporting Software 1-6

ISAAC Installation 1-6

Foxpro Version 3.0 Installation 1-7

ISAAC Demonstration and Approach 1-7

Selecting Model Input 1-8

Model Approach 1-12

Model Execution 1-16

Model Output 1-18

The Structure of the User's Manual 1-20

Chapter 2. Basic Requirements Model Runs 2-1

Introduction 2-1

System Requirements 2-1

Model Operation 2-3

Basic Requirements Run Sequence 2-3

Model Input 2-4

Parameters 2-4

Lower Box 2-5

Upper Box 2-9

Middle Box 2-10
Model Output 2-14
Summary Output 2-14
Viewing Other ISAAC Reports 2-20

Chapter 3. Advanced Requirements Model Runs 3-1

Stock Options 3-2
Include Starting Assets? 3-3
Use Pre-specified Buy Quantity ? 3-6
Force Buy Based on Pipeline % Below 3-12
Resupply 3-13
Day Base or Depot Repair Begins 3-13
Day Order and Ship Begins 3-14
Number of Warning Days 3-14
Other Options 3-14
Exponential Repair 3-14
Variance to Mean Ratio 3-15
Number of Bases 3-15
Optimization 3-15

Chapter 4. Developing Input Kit Data 4-1

Structure of Chapter 4 4-3
Kit Pull-Down Menu Options 4-3
New Kit 4-3
Copy Baseline Kit 4-6
Export Kit 4-8

Import Kit 4-10

Common Components 4-12

Steps for Importing a Kit 4-12

View \ Edit Existing Baseline Kit 4-18

Sensitivity Changes 4-19

Kit Parameters 4-20

Bottom Row of Buttons on the Kit Parameters Screen 4-21

Top Section of Kit Parameters Screen 4-22

Middle Section of Kit Parameters Screen 4-22

Lower Section of Kit Parameters Screen 4-23

Component Data 4-23

Identification Fields 4-24

Basic Fields 4-25

Conditional (Wartime / Non-wartime) Fields 4-28

Viewing Different Components via the Component List 4-30

How to Create New or Modify Existing
Component Information 4-31

Creating New Component Records 4-31

Adding New Component Records Using
the Default Fields 4-32

Copying Existing Component Field
Information into a New Component
Record 4-32

Editing Existing Component Records 4-33

Deleting Existing Component Records 4-33

Deleting One Record at a Time 4-33

Deleting Multiple Component Records 4-33

Warnings that Occur During Component Record Editing or Creation 4-34

Warnings Generated by Invalid Field Values 4-35

Validation for Component Fields 4-35

Developing the Component Indenture Structure 4-37

Common SRU Components 4-37

Errors and Warnings from Creating the Indenture Levels 4-39

Global Changes (Sensitivity) to Kit 4-42

Basic Global Changes Sequence 4-42

Detailed Global Changes 4-43

Global Changes with Filters 4-47

Miscellaneous 4-49

Chapter 5. Input, Output and Miscellaneous Data Windows 5-1

Browse 5-2

Moving the Window 5-2

Resizing the Window 5-2

Splitting the Window 5-2

Changing Displayed Field Size 5-3

Moving Fields 5-4

Browse Pull-down Menu 5-4

Change 5-4

Grid 5-4

Link Partitions 5-5

Change Partition 5-5

Size Field 5-5

Move Field 5-5

Resize Partitions 5-5

Goto... 5-6

Seek... 5-6

Toggle Delete 5-7

Append Record 5-7

Printing Input and Output Field Data 5-7

Indexed Browse Windows 5-9

Applicable Windows 5-9

Sequence of Operation 5-10

General Characteristics 5-10

Temporary Databases 5-11

Input Data Windows 5-11

Component Data 5-12

Run Log 5-13

Output Data Windows 5-13

Pipeline Data 5-14

Shopping List Data 5-15

Curve 5-18

Yearly Cost 5-20

Critical Item List 5-21

Critical Item Ranking 5-21

Critical Items Under Cannibalization 5-22

Critical Items with No Cannibalization 5-22

Critical Items with Partial Cannibalization 5-22

Common SRU Components 5-23

Critical Item Browse 5-23
Field Groupings 5-24
Field Views and Indexes 5-25
Miscellaneous Data Windows 5-28
Statistics 5-29
View Input-Output 5-29
View Shop Comparison 5-31
Chapter 6. Text Editor and IPSS Interface 6-1
Introduction 6-1
The ISAAC Text Editor 6-1
Opening and Closing Files 6-3
Opening Files Using the ISAAC Open Dialog Box 6-3
Edit Preferences 6-4
Closing Files 6-4
Text Options 6-5
ISAAC to IPSS Interface 6-7
Relationship between ISAAC and the IPSS 6-7
Preparing ISAAC Output for Export to the IPSS 6-7
Data that is Exported to IPSS 6-9
Model Run Summary Data 6-9
Item Level Model Output Data 6-10
Chapter 7. Programs that must be Executed Outside of ISAAC 7-1
Introduction 7-1
Filer For ISAAC 7-1

ISAAC Filer Options 7-1
Archive 7-1
Retrieve 7-2
Using the ISAAC Filer 7-2
Set Keys 7-4
Using Set Keys 7-4
Set Keys in Operation 7-4
Installation 7-5

Chapter 8. Evaluating Spares Mixes with ISAAC 8-1

Introduction 8-1
Special Purpose Evaluation for Assessments 8-1
Chapter Outline 8-2
Basic Evaluation Run Sequence 8-2
Implementation Details 8-3
Select Spares Mix 8-4
Use Spares Mix 8-5
Permit More Buys? 8-5
Multi-day Evaluations 8-8
Implementation 8-9
Multi-day Browse 8-10
Stock Option Impacts on Evaluation 8-14
Starting Assets 8-14
Prespecified Buy Quantities 8-15
Improving Forecast of Wartime Demand 8-16

Appendix A. Glossary of Screen Terms

Appendix B. Data Dictionary

CHAPTER 1

Introduction to ISAAC

PREFACE

Initial provisioning occurs whenever the Israel Air Force (IAF) buys new weapon systems, whether it is procuring an additional squadron of F-16s or a few helicopters not currently in the IAF. Along with the purchase of aircraft, the IAF acquires spare parts (spares) in case parts fail. The purpose of the spares procured in conjunction with the aircraft is to support the new weapon system over some period until the standard spares replenishment system can begin to provide support.

During initial provisioning, the IAF must address three basic questions: (1) which spares should be ordered within a specific period, given a constrained budget; (2) when should they be ordered; and (3) how should funds from the total budget be obligated over time? The Initial Spares Aircraft Availability Calculation (ISAAC) was designed to help the IAF initial procurement team answer those questions. The model can estimate spares requirements by using an optimization method that considers a number of critical factors (failure rates of components, projected war and non-war scenarios, alternative maintenance strategies, etc.). The model user can vary those factors and thus better understand the impacts of critical decisions. This user's manual explains the use of the ISAAC Version 3 model, tells how to operate it, describes the basic model assumptions, defines the input and output, and presents examples of possible analyses.

The manual should be helpful to all IAF logistics analysts with access to the ISAAC model who are responsible for initial provisioning of spares requirements or policy. It describes the steps required to run the personal computer (PC) version of the model. It covers the model's basic purpose, its installation procedures, step-by-step instructions for basic and advanced model operations, input parameter guidance, and a sample demonstration.

OVERVIEW OF ISAAC

ISAAC employs a weapon-system approach in selecting specific spare parts to be procured as part of initial provisioning. This approach calls for item decisions to be based explicitly on their effect on the weapon systems being supported. As the measure of performance, we use the ultimate goal of the IAF logistics system: available aircraft. The availability rate (at a particular time for a particular aircraft type) is the percentage of the fleet that is not grounded for lack of spares. Reparable spares and consumable items — line and shop replaceable units (LRUs and SRUs) — are bought using a marginal analysis technique that purchases items on the basis of their contribution to weapon-system availability (benefit) per unit cost. The potential buys are ranked in order of decreasing benefit per unit cost. Buys are then made in that order until a budget target or availability target is reached. This marginal analysis approach can be applied to any weapon system, from aircraft to ground support radars.

Factors Considered by the Weapon-System Approach

The major factors considered by our basic weapon-system approach are aircraft availability and the cost of spares required to achieve a given number of available aircraft.

- We define aircraft availability as the percentage of aircraft not inoperable because of lack of a spare. This definition of availability looks only at the supply of spares — it does not consider such actions as on-aircraft maintenance (scheduled or unscheduled) or the effect of shortages of repair parts such as nuts and bolts. It also does not consider exceptional actions such as expedited repair or procurement.

- We use the term "buy cost" to refer to the purchase cost of spares required to achieve a specific number of available aircraft. The buy cost considers only the actual spares' purchase cost to the supply system — it does not consider maintenance or administrative costs.

Aside from availability and budgets, this approach considers a number of other factors:

- *Essentiality*. The model focuses on the items that affect availability and cost the most — the "essential" items. These are the high-cost, high-indenture items (i.e., reparable LRUs, reparable SRUs, and expensive consumables). Including all IAF items would make the model cumbersome and difficult to implement and would yield little improvement in availability.

- *Indenture:* Each aircraft has an indenture structure. Aircraft are composed of LRUs, LRUs are composed of SRUs, SRUs are composed of lower indenture SRUs, and so on. The model develops the optimal balances between procuring LRU and SRU spares. A more expensive spare LRU has a greater impact on availability (when a spare LRU is available, the aircraft is operational almost immediately), while the less expensive SRU affects availability only indirectly (even if an SRU spare is immediately available, the aircraft must wait for the LRU to return from maintenance).

– *War and non-war conditions.* The model estimates spares requirements under the two very different operating environments of wartime and non-wartime conditions. Wartime scenarios are characterized by dynamic flying hour profiles and by maintenance philosophies that differ radically from those associated with non-war scenarios.

– *Common components.* Common components are components common to more than one aircraft series or type, such as components on the aircraft series F-16D, F-15C, and F-15I. Since the weapon-system approach estimates availability by aircraft series, the method must also consider the economies of scale gained for components common to many aircraft (i.e., when these economies of scale are taken into account, the total spares requirement is less than it would be if each weapon system were considered as separate and independent).

– *Cannibalization*. The ability to consolidate broken LRUs onto a single aircraft greatly improves aircraft availability without increasing procurement costs. For instance, if one aircraft is missing item A and another is missing item B, maintenance can remove an operational item B from the first aircraft and install it on the second, returning that aircraft to service. Cannibalization yields improved aircraft availability over the non-cannibalization case for the same procurement cost. The disadvantage of cannibalization for the maintenance system is that flight-line maintenance actions and expected back orders (EBOs) are increased.

– *Component stock considerations.* The model incorporates pre-specified user spares decisions to handle situations in which assets exist or orders have been previously placed.

– *Other item factors.* The model also considers a number of other item-specific factors such as demand (failure) rates per flying hour, base and depot repair times, condemnation rates, transportation times, unit cost, quantity per application (QPA), and procurement lead-time total (PLTT).

By considering all these factors, the model produces a robust spares requirement that determines the spares that will be required during a user-specified time or "coverage period," typically several years. The model also estimates the appropriate time to order those spares and the costs incurred during ordering.

ISAAC can be used to determine spares requirements and also to evaluate the availability yielded by a predetermined spares mix. The requirements capability of ISAAC enables the user to enter a target (cost or availability), and then the model calculates the spares mix required to meet that target. ISAAC's evaluation capability approaches the spares mix problem from the opposite direction — ISAAC enables the user to select a spares mix, and then it calculates the availability that mix provides. The evaluation mode is usually used to determine how well a given spares mix performs under conditions that are different from the conditions of the original requirements run.

Capabilities Provided by the ISAAC Interface

The ISAAC interface combines high-speed, weapon-system-oriented spares computations with the database processing power of Microsoft FoxPro. ISAAC provides spares procurement analysts with the following capabilities:

Library of previous runs (input and output) and archival/retrieval capability. ISAAC maintains a library of previous runs that can be used for comparative analysis purposes. The archiving / retrieval capability gives the analyst a way to easily store and retrieve spares analyses (model output) along with related item-level information (model input) in user-specified locations.

Sensitivity analysis capability (global change). ISAAC lets the user easily analyze the impact of input data changes on model output results. One can easily alter all or some of the item input. For instance, the user can increase all demand rates by 10 percent or increase only those demand rates for items with costs greater than $200,000.

Ease of analysis. Through its database interface, ISAAC allows the user to easily view weapon system databases with large numbers of items. For instance, one can select the data fields to view by checking them from an inclusive list. The model then displays and prints only those selected fields, in a tabular form. When any item field in the table is clicked on, the model sorts the data so that the user can immediately identify the worst-performing items.

Comparative analysis. ISAAC allows the user to compare any two model-generated spares requirements for the same weapon system. By simply selecting the two previous model results, the user causes the model to present a side-by-side comparison of the spares requirements and to sort the information so the items with the biggest differences are at the top of the list.

INSTALLING ISAAC

Conventions

Before describing the model installation process, we will discuss some of the basic conventions (mostly standard Windows conventions) we use to describe the model, the model commands, and how to initiate model processing. The rest of this chapter will address the user directly.

– Screens within ISAAC enable you to modify (edit) data or to initiate model processes. You cannot modify the presentation of an ISAAC screen, but much of the information on the screen can be modified in various ways:

  ► Text boxes let you type information in them. When you move to an empty text box, an insertion point (a flashing vertical bar) appears. The text you type starts at the insertion point. You can also delete the existing text by pressing **'DEL'** or **'Backspace'**. [1]

  ► Drop-down list boxes appear initially as a rectangular box containing the current selection. When you select the down arrow in the square box at the right, a list of available choices appears. If there are more items than can fit in a box, scroll bars are provided.[2]

– Windows within ISAAC allow you to view data in tabular form, usually without your being able to edit the data. Most of the windows in ISAAC are browse windows that enable the user to modify the presentation of the data (number of fields, sequence of fields, and appearance of the field data). For instance, to let you select the fields you need to view in the browse window, the model lists all possible fields on several "picklists." Check the box to the left of each field you want to view. Some of the abbreviated field names are probably familiar to you, while others are unique to ISAAC. All of the abbreviated field names are defined in Appendix B — Data Dictionary — in alphabetical order.

– "*Click* on a button" means move the mouse cursor to a position on top of the button on the screen and push the mouse's left button.

– "Choose" refers to selecting choices from the menu bar and pull-down menus. Specifically, "choose" means place the mouse cursor on top of the specified selection and click on it. In general, you will first choose an item on the menu bar (thus opening a pull-down menu), and then you will choose a selection from the pull-down menu.

– **Bold letters** represent computer commands that you click on or type in. **'Single quotation marks'** around one or more letters indicate that it or they are a valid entry for the respective field, or else that you may enter a keyboard command such as **'Esc'** or **'Tab'**.

– To view information not currently displayed on the screen — scrolling — press the left or right arrow key to scroll horizontally and the up or down arrow key to scroll vertically.

– All screen buttons have two shades for their commands — dark-shaded letters indicate that the user can access the button's operations, while light-

[1] "Text boxes" has been extracted, with minor modifications, from Microsoft Windows User's Guide (Version 3.1),1992, p. 30.

[2] "Drop-down list boxes" has been copied from Microsoft Windows User's Guide (Version 3.1),1992, p. 31.

shaded letters indicate that the button is not operational given the current conditions.

– The **'Esc'** key can be used to revert to the previous screen. It can generally be used as an "undo" command to negate a previous entry.

– The fields are not case-sensitive (i.e., an entry of **'Y'** is the same as an entry of **'y'**).

Loading ISAAC and Supporting Software

Loading ISAAC onto your hard drive requires the installing two groups of disks: ISAAC application file installation disks and Microsoft Foxpro 3.0 installation disks.

ISAAC Installation

The "install" program transfers the ISAAC application files to your hard disk. To start the installation:

1. Start Windows and open the Program Manager.

2. Insert ISAAC Disk #1 of 4 in drive A and close the drive door.

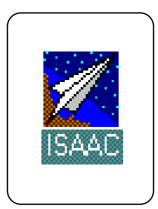
3. Choose **File** from the menu bar.

4. Choose **Run** from the File pull-down menu.

5. Type **A:SETUP** and choose **OK**.

6. Choose the drive and windows folder in which you wish to place the ISAAC files (the installation disks take a total of about 15 minutes to install).

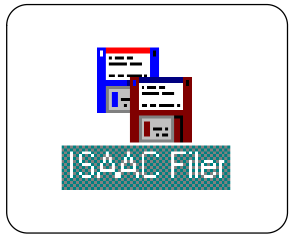
In addition to the ISAAC icon,



you will see two other icons in this windows folder:

a **Set Keys** icon

and an **ISAAC Filer** icon



If you do not see the **Set Keys** and **Filer** icons, go to Chapter 7 for instructions on their installation. Those two icons access programs that interact with ISAAC but that must be run outside of ISAAC. Both of these programs will be described in Chapter 7 — Programs That Must Be Executed Outside of ISAAC.

Foxpro Version 3.0 Installation

The "install" program transfers the Foxpro 3.0 files to your hard disk. To start the installation:

1. Start Windows and open the Program Manager.

2. Insert Foxpro Version 3.0 Install Disk #1 in drive A and close the drive door.

3. Choose **File** from the menu bar.

4. Choose **Run** from the File pull-down menu.

5. Type **A:SETUP** and choose **OK**.

Choose the drive and windows folder in which you placed the ISAAC files (the installation disks take a total of about 15 minutes to install).

ISAAC Demonstration and Approach

This section presents a simple demonstration (a model run) of ISAAC and describes the basic model approach to developing spares requirements. We use the term "model run" to mean a complete cycle of model processing, from input to output. The demonstration focuses on four components: starting the model and selecting the required model input, describing the basic model approach, describing how to execute a specific model run, and finally viewing the model output. You can actually run the model as we step through the demonstration (specific user actions are indicated by the diamonds), or you can simply look at the document figures that show the model screens.

Selecting Model Input

In the previous section you installed the model. In this section we describe how to run it.

– You start by opening the Windows Program Manager and using the mouse to double-click on the **ISAAC** icon. Note: you should close all other applications before running the model, in order to ensure that sufficient memory is available.

The main menu (top line of your screen) lets you select option **Model**, **Kit**, **Text Editor**, or **Exit** (see Figure 1-1). The options **View Input**, **View Output**, **Misc**, **IPSS**, are deselected until after you select the **Model** option.

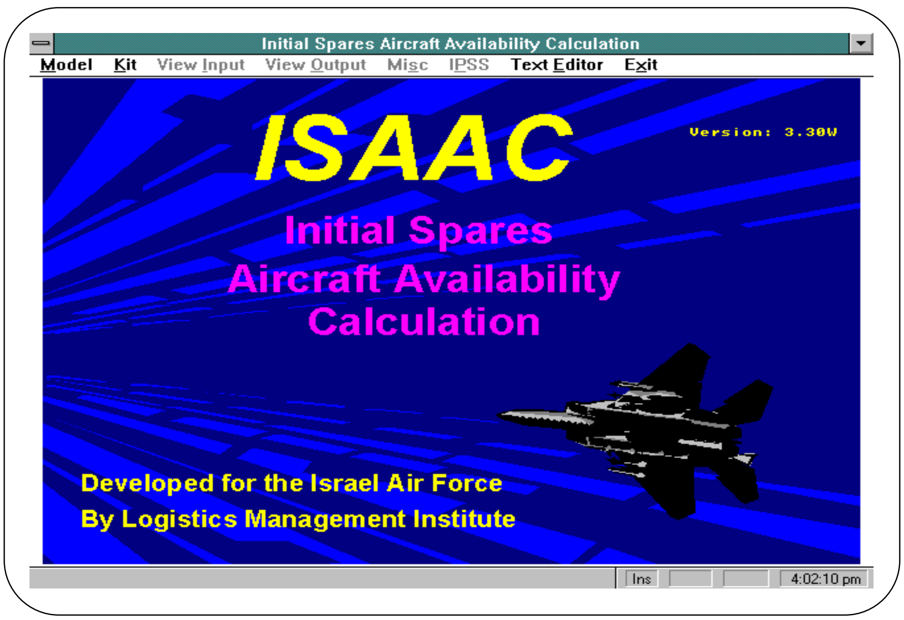


Figure 1-1
ISAAC Initial Spares Aircraft Availability Calculation

– The main menu items, described here, are discussed in detail elsewhere in this user's manual:

► The **Model** option enables you to make model runs. The basic model options are described in Chapter 2; the advanced options are described in Chapter 3.

- The **Kit** option focuses on developing all of the required model input such as the component data (failure rates, repair times, etc.) and global parameters (availability targets, maintenance philosophy, etc.). It is described in Chapter 4.

- **The View Input**, **View Output,** and **Misc** options enable you to view component-level and model-run-level input and output. These options are described in Chapter 5.

- The **IPSS** (Initial Provisioning Support System) interface and the ISAAC **Text Editor** options are described in Chapter 6. The IPSS interface allows you to export ISAAC model run output to the IPSS. The text editor provides basic text editing capability.

- **Exit** will close ISAAC.

– The **Model** option lets you start a model run — our focus in this demonstration. Choose **Model** from the menu bar.

The screen in Figure 1-2 should appear on your monitor. You are now at the Model Parameters Screen, where you can see the basic input for the most recent run of the model. The user interface has two basic purposes: (1) to allow you to run the model with minimal difficulty, and (2) to keep track of your runs and allow easy access to a library of previous runs' inputs and outputs. For most model runs, it is easier to modify a previous set of run parameters than to start from scratch and modify many inputs. We will first show how to view a previous run's input and output. Then we will show how to modify a previous run and create a new run. Each step in the demonstration is described in more detail in Chapter 2.

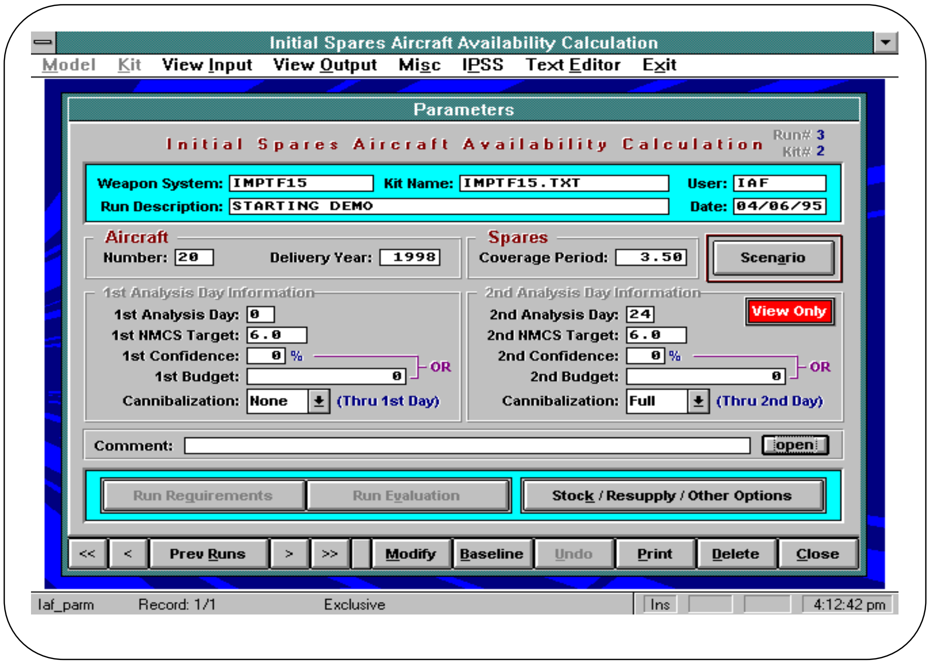


Figure 1-2
Model Parameters Screen

– To choose from the library of previous runs, click on the **Prev Runs** button (bottom of the window) now. (Note that when you enter this window, the currently displayed parameters are from the last model run you made. If that is the run you want to view, no selection is necessary.) If you have made no model runs, then only our start-up run is in the library now.

– You are now viewing a window that displays the library of past runs. Move the mouse cursor to the run labeled **Starting Demo —** and click on the left mouse button to highlight that run. Then click on the right mouse button to select that run.

You are now back to the parameter screen, where you can see the basic input for the run you selected called **Starting Demo**.

– You can view the detailed input or output data associated with this run by clicking on **View Input** or **View Output** from the main menu. The model will not let you modify any information while it is in the **View Input** or **View Output** Window. Click on **View Input**.

– To view the component data (LRU and SRU information), click on **Compo-nent Data** now. The window in Figure 1-3 should appear on your screen.

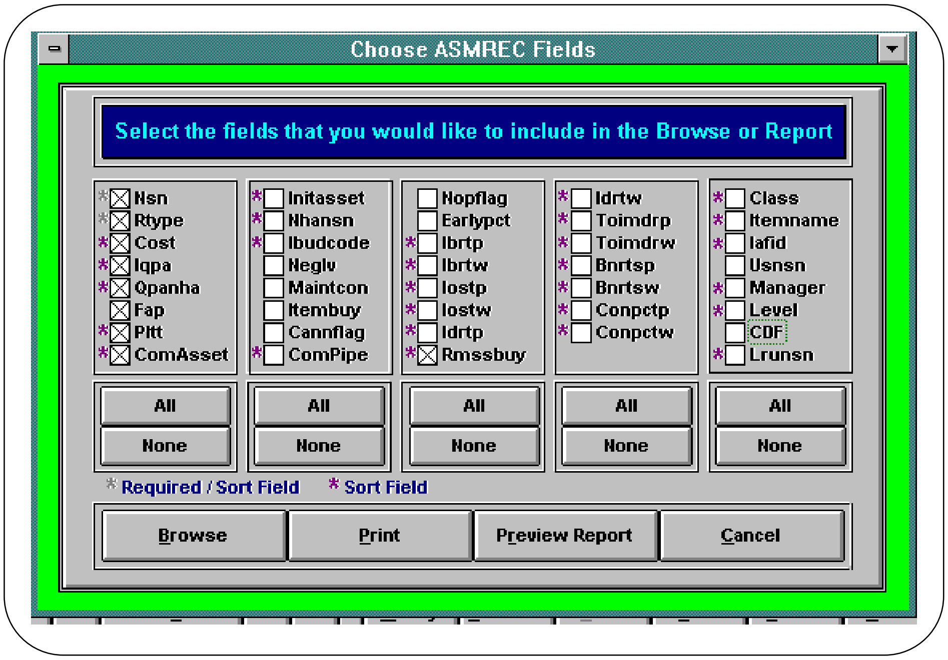


Figure 1-3
Component Data Picklist

You are now viewing a “picklist” window showing all the component data fields. To view individual data fields, ensure that an ‘x’ is in the check box to the left of the field name.

– Clicking on the check box will toggle the selection on and off for viewing the field. A description of each of those fields, in alphabetical order, is in Appendix B. When you have picked the fields you want to view, click on **Browse**.

You are now viewing the component database for the run labeled Starting Demo. (The interface allows you to view any other previous run’s information in a similar manner via the **Prev Runs** button — at the bottom of your screen.) (You may move the view window by dragging the title bar. You may scroll left, right, up, or down by using the arrow keys. Pressing the arrow keys lets you scroll to view other fields and records.)

– When you are through viewing the component data, double-click on the upper left-hand box to close the window.

You are now back to the Model Parameters Screen (Figure 1-2). The middle section of this screen contains the primary global parameters (Figure 1-4). In the Model Approach sub-section, following, we explain the purpose and meaning of each key parameter.

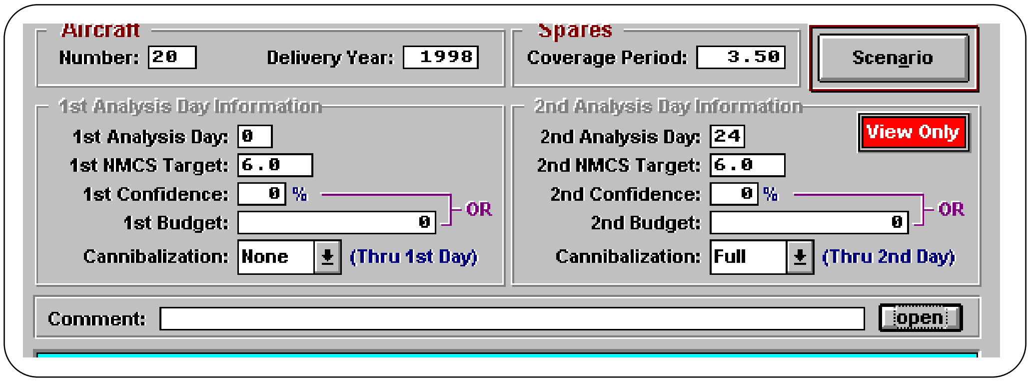


Figure 1-4.
Primary Global Parameters (Middle Section of the Model Parameters Screen)

Model Approach

ISAAC employs a weapon-system approach in selecting spare parts to be procured as part of initial provisioning. Reparable spares and critical consumable items are "bought" using a marginal analysis technique that identifies LRUs and SRUs on the basis of their contribution to weapon-system availability per unit cost. Spares that provide the greatest availability per unit cost are selected before those that yield less availability per unit cost.

Figure 1-5 presents an example of the basic model approach for estimating spares procurement. The figure displays two key periods:

- Coverage period — ISAAC estimates the spares required from the time the first aircraft arrives to the time the standard replenishment system can begin to provide spares support. We define that period as the coverage period.

- Ordering and budget period — ISAAC estimates when the IAF should buy the spares in order to have them arrive in time. To place orders, the IAF must budget dollars on the basis of when the order is placed. An annual procurement equals the cost of all orders placed in a given year (the buy cost).

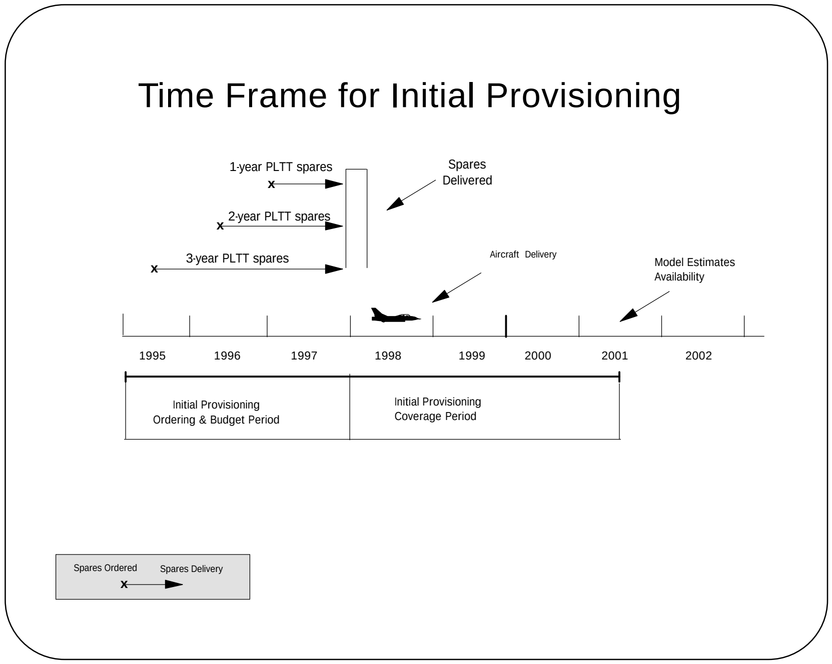


Figure 1-5
Time Frame for Initial Provisioning

You specify both of these periods by entering the length of the coverage period and the year of the aircraft delivery in the middle section of the Model Parameters Screen (Figure 1-4).

For instance, in Figure 1-5, we assumed a coverage period of 3.5 years (the IAF states that assumption as a 3-year coverage period starting halfway through the first year of delivery) and an aircraft delivery of 1998. With that information, the model starts the coverage period on the first day of the year the aircraft are delivered (1998) and ends the coverage period at 3.5 years later, or midway through 2001. The model determines the ordering and budget period using the aircraft delivery year and each item's PLTT. The model assumes that all initial provisioning spares will arrive in the aircraft delivery year and that their impact will be felt one PLTT in advance of their delivery (i.e., at the time the order is placed). Thus, the start of the ordering and budget period occurs a maximum PLTT before the aircraft delivery, and it ends the year before aircraft delivery. For our example, if the longest PLTT of all the items is three years, the model assumes that the ordering and budget period begins in 1995 and ends in 1997.

Essentially, the IAF wants to estimate requirements for a spares mix robust enough to accommodate two very different sets of conditions — non-war and war. For non-war situations, IAF maintenance assumes that it will not cannibalize LRUs. But in wartime, cannibalization is allowed for some LRUs, to increase aircraft availability. Under non-wartime conditions, the IAF assumes that weapon-system flying hours will be stable over the coverage period; in wartime, they may vary significantly on a daily basis. Under non-war conditions, repair turnaround times may be on the order of months; in war they may decrease for items repaired in Israel but drastically increase for items repaired outside of Israel. While we will wait until Chapter 4 to treat the item-specific assumptions that vary between war and non-war conditions, we will discuss some of the global assumptions now.

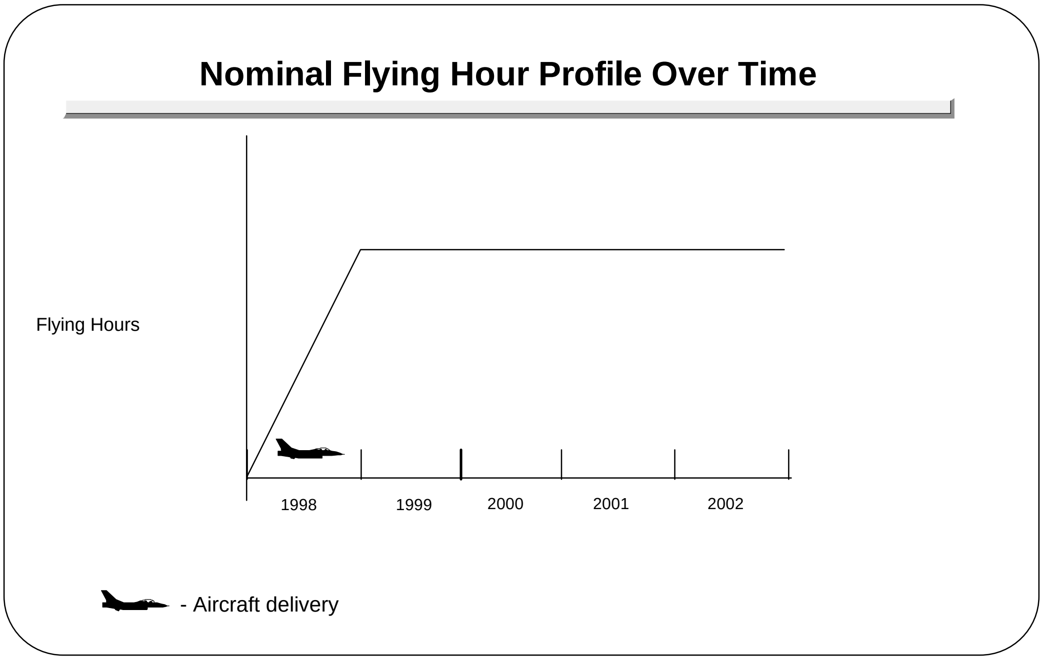


Figure 1-6
Nominal Flying Hour Profile Over Time

Figure 1-6 plots a nominal example of the total flying hours for a fleet of aircraft for a non-war scenario. In this example, the only year in which flying hours change is the delivery year; this is because not all of the aircraft are delivered simultaneously. We make a key simplifying assumption that aircraft delivery is spread out evenly over one year. Thus, the first year has only half as many flying hours as do each of the following years. The result is that the number of full years of flying is the coverage period less a half a year or, for our example, three years of steady-state conditions (what we will term a "steady-state coverage period").

The model procures spares for the steady-state conditions just described as well as for a war that can last up to 99 days. We assume that the war scenario occurs at the end of the steady-state coverage period — the most demanding scenario from a spares requirements standpoint. The war scenario also assumes dynamic flying hours, which the user specifies by day.

– You can specify the war and non-war flying hour scenarios by clicking on the **Scenario** button on the Model Parameters Screen (Figure 1-4).

As we have said, the ISAAC model is designed to estimate spares requirements to support both wartime and non-wartime scenarios. While ISAAC refers to "analysis days" (on the Model Parameters Screen), this does not mean that those two days worth of spares are bought. Each analysis day represents a cumulative period up to and including the respective analysis day. A “0” for the model’s analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A “30” translates into having the model procure spares for a 30-day war scenario that starts at the end of the coverage period and ends 30 days later (alternatively, any other day of the wartime scenario could be entered). Thus, analysis days are days of interest. While the basic operation generally uses a day of non-war and the last day of the war, the model is flexible and can operate on any one or two days of interest.

Figure 1-7 illustrates flying hour profiles over a 30-day war for three different weapon systems (aircraft) —” A,“ ”B,“ and ”C.“

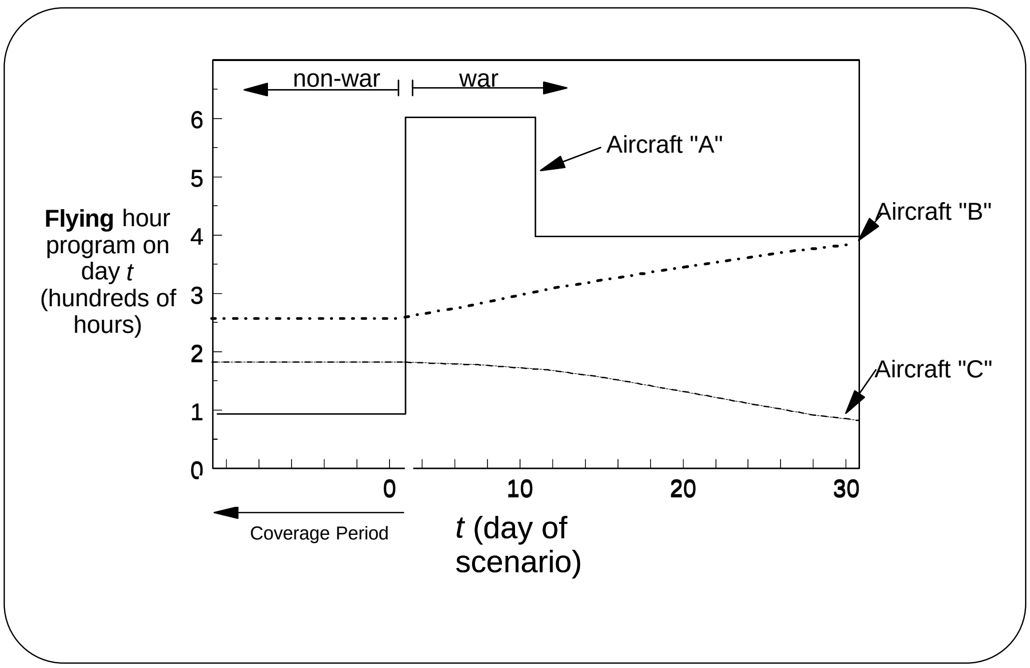


Figure 1-7
Sample Flying Hour Profile for Three Weapon Systems

To be most useful, any day to be analyzed should fall within the time during which the supply system and the repair system are the most stressed. In the wartime scenario, that day would usually coincide with the end of the war, with flying hours increasing (as is the case with aircraft ”B“ Figure 1-7). But, if the flying hour scenario looks like that of aircraft ”A,“ then you should analyze Day

11, since doing so probably results in generating the greatest spares requirements of the wartime scenario.

Conceptually, the model "purchases" spares in two modes of operation: either to meet an availability target (unconstrained) or to meet an availability target under a budget constraint. The availability target is expressed in terms of NMCS (not mission capable for supply) or AOG (aircraft on ground) aircraft. This can be expressed mathematically as (1-NMCS/#Aircraft), where #Aircraft equals the number of aircraft under consideration. ISAAC will always attempt to meet the availability target(s) for the least cost. In the unconstrained budget mode, the availability targets should always be met. In the budget-constrained mode, the model may not meet the availability target on either or both days, but neither will it exceed the budget constraint.

The user enters the availability or budget target for each analysis day and runs the model. For the first analysis day, the model purchases enough spares to meet the target or — if constrained by the budget — purchases the spares that yield the best spares mix without exceeding the budget constraint. For the second analysis day, the model considers those spares bought in support of the first day as being in the inventory (the model also considers the changes the maintenance system undergoes while making the transition from non-war to war conditions). It then purchases additional spares to meet the specified second-day target or second-day incremental budget constraint (if applicable). Thus, the spares that the model selects for the second analysis day make up the difference between the first day's buy and those required to meet the second target. After processing both days' parameters, the model returns to the first analysis day and recalculates performance with the total quantity (a shopping list) of spares purchased to support Day 1 and Day 2. With that model operation, the first analysis day usually does better than the second as far as meeting its availability target is concerned. That is because the model has purchased enough spares to meet the first day's target in addition to purchasing the spares required to meet the second day's target.

Thus, the model can consider in its spares calculations steady-state conditions (non-war scenario), dynamic conditions (the war scenario), and the transition between the two.

Model Execution

Next we will complete model input preparation and execute a spares requirements model run using ISAAC. Then we will modify the selected run and rerun it:

– To do that, click on the **Modify** button (bottom row on your screen [see Figure 1-2]). Once you click on **Modify**, the run description becomes blank. You can no longer select **Prev Runs** to view other runs or select **Baseline** (though all other fields remain the same). Selecting **Modify** informs ISAAC that you are going to modify one or more model parameters and produce a new run. This requires a new run description in order to distinguish this new run from previous runs.

– The model moves you to the **Run Description** field, and then you type in a new description (e.g., ISAAC DEMO II). Make sure to click on **Enter**“ before you move off the **Run Description** field. The only other identification field value that you can change is the **User** field. Notice that, as soon as you finish entering the run description, the **Run Requirements** button shading darkens, indicating that you have moved from just viewing the data to being able to run the model.

– Next, you can change any of the primary global parameters you have been viewing in the middle box of Figure 1-2; (that is, Figure 1-4). Those values are your basic targets (NMCS, the confidence level in achieving that NMCS, or the budget) and operating conditions (full, partial, or no cannibalization) for the first and second analysis days.

– For now, increase your NMCS (the AOG value) targets by one for **1st NMCS Target** by clicking on the box to the right of the field name and typing in the number.

– Now click on the **Run Requirements** (see Figure 1-2) button to run the model. A window (Figure 1-8) will open, displaying the number of LRUs and SRUs being processed.

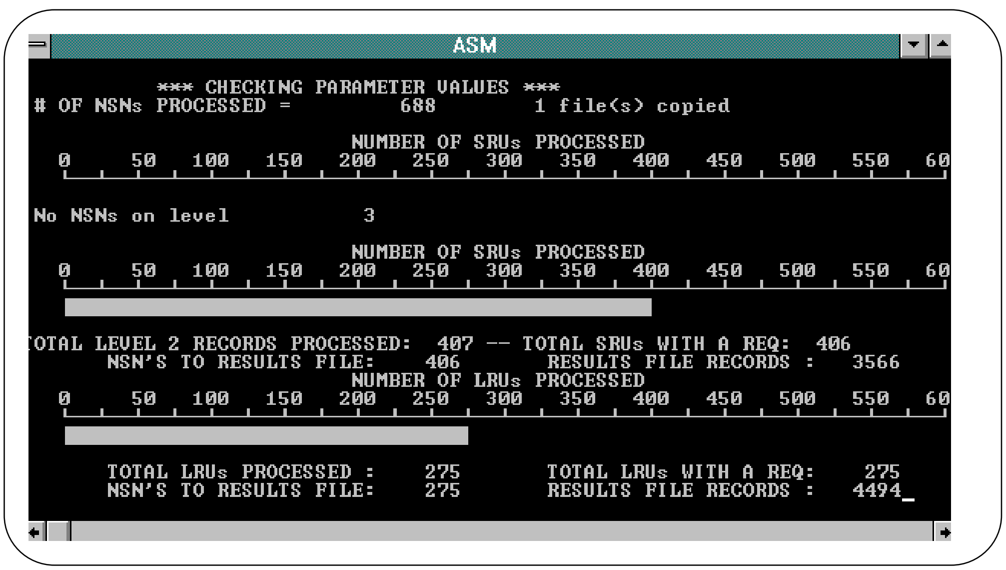


Figure 1-8
ASM (Aircraft Sustainability Model) Window

Once processing has been completed, this window will close automatically and a Processing Model Output Window will appear. This run will take a few minutes.

Model Output

Once the model has developed spares requirements for war and non-war conditions (what we term a spares shopping list for the two analysis days), it can then determine when to order the spares and the resulting estimated costs by year. That information is displayed via the screen titled Spares Cost Summary a PLTT before Spares Delivery (Figure 1-9). To estimate that information, the model starts by assuming that the manufacturer delivers all spares on the first day of the delivery year (e.g., the first day of 1998). Then, by moving an item's PLTT back in time (the PLTT is given data), the model estimates the year the order needs to be placed (see Figure 1-5). The model can also estimate the annual buy cost (see Figure 1-9). For instance, all spares with less than a 1-year PLTT are ordered in 1997 and added to the 1997 spares buy cost, all spares with a 1-to-2 year PLTT are ordered in 1996 and added to the 1996 spares buy cost, and so on.

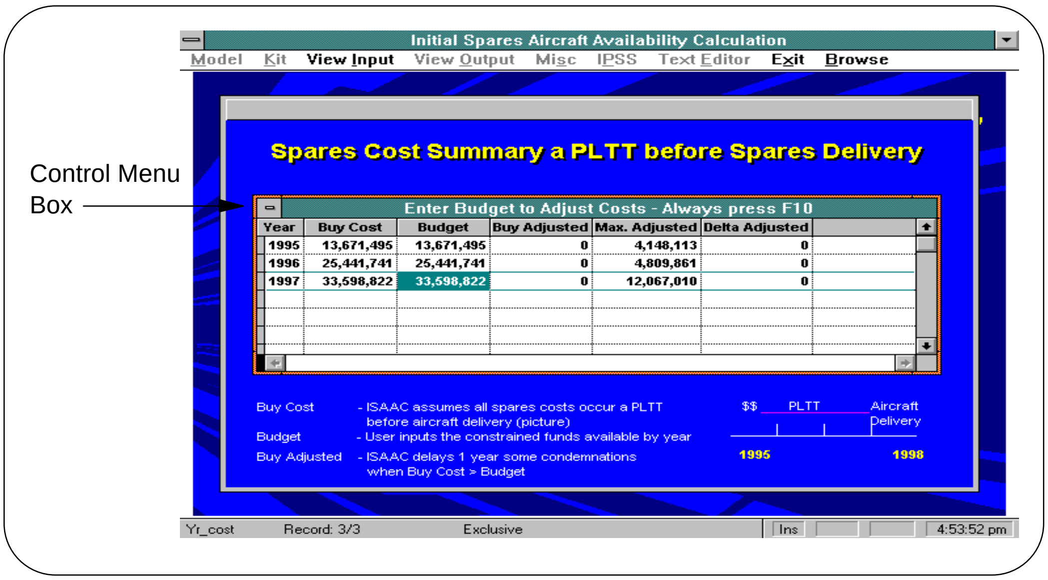


Figure 1-9
Spares Cost Summary a PLTT before Spares Delivery Screen

An additional consideration is that the IAF budget constraint has two parts: a total constraint (e.g., $100 million) for all initial provisioning spares, and an annual constraint for each year (e.g., $20 million, $30 million, and $50 million for 1995, 1996, and 1997, respectively). You can enter the annual constraints in the budget column of Figure 1-9, and the model will then delay less important orders to try to accommodate those constraints. We will describe that operation in Chapter 2.

– Press the **F10** function key, causing cost-adjusted values to be calculated for each year of the coverage period. Then close the window by clicking on the Control-menu box (see Figure 1-9).

The Processing Model Output Window will reappear while the model completes its output processing. When the model has finished, it will generate a report summarizing cost and performance statistics for the two analysis days (see Figure 1-10).

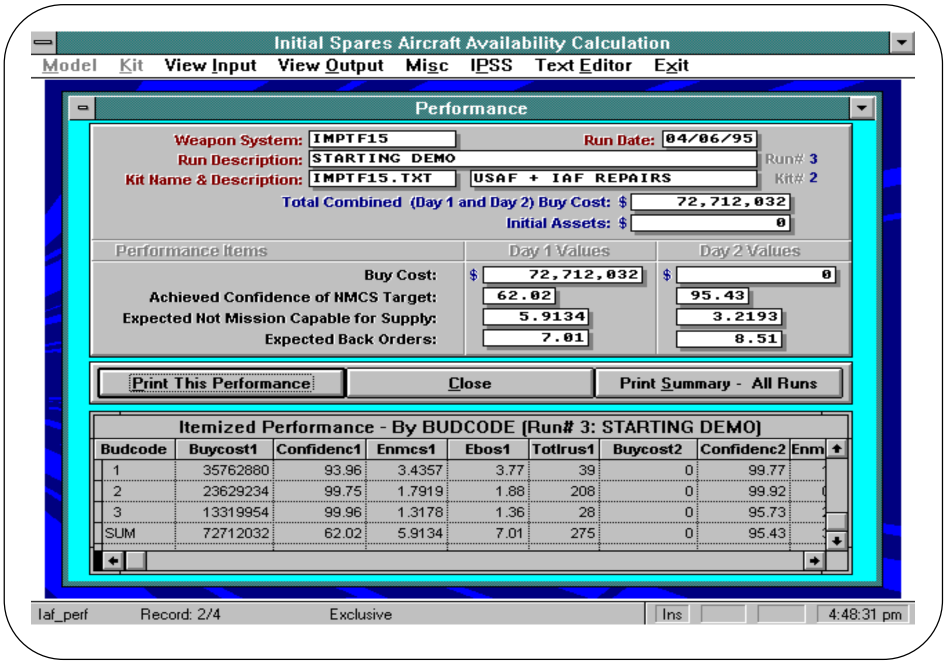


Figure 1-10
Performance Report Window

– Click on the **Close** button in the middle of the screen to continue and then choose **Model** from the menu bar.

You are now back at the starting ISAAC screen (Figure 1-1). Click on **Model** and the Model Parameters Screen you just modified appears. At this point you may click on **View Output** to display the possible output reports. As with the **View Input** reports, once you select a report, you can then select the specific fields to view by clicking on the corresponding check boxes from a pick-list. The **Shopping List Data** report lets you view the model-recommended spares buy (BUY_TOTAL) and other related fields described in Chapter 5.

THE STRUCTURE OF THE USER'S MANUAL

This document is structured for ISAAC model users interested in various levels of detail. It consists of eight chapters and two appendices.

Chapter 1 — Introduction to ISAAC — presents procedures for installing ISAAC on a PC and step-by-step instructions for demonstrating ISAAC's capabilities.

Chapter 2 — Basic Requirements Model Runs — presents system requirements, basic model screens, definition of and guidance for basic user inputs, and an explanation of basic model output.

Chapter 3 — Advanced Requirements Model Runs — presents the advanced user model screens along with the definition of and guidance for advanced user inputs, as well as an explanation of advanced model output.

Chapter 4 — Developing Input Kit Data — presents the "kit" screens, defines the parameters that are unique to the kit, and gives guidance for using the kit screens to develop or import the item-level and global input required to run the model.

Chapter 5 — Input, Output, and Miscellaneous Data Windows — provides information on how to browse and print model input and output data and describes the model input and output data windows.

Chapter 6 — Text Editor and IPSS Interface — describes the ISAAC Text Editor and the export of ISAAC's shopping list to the IPSS model, which develops the actual order forms.

Chapter 7 — Programs That Must Be Executed Outside of ISAAC — describes how to archive and retrieve sets of ISAAC runs and other related utilities.

Finally, Chapter 8 — Evaluating Spares Mixes with ISAAC — describes how the user can select a particular spares mix and then use ISAAC to calculate the availability that mix provides. Since that chapter draws on information presented in others, we place it last.

The two appendices are organized as follows: Appendix A — Glossary of Screen Terms — briefly discusses each descriptor presented on the various ISAAC screens (i.e., [Kit] Parameters; [Model] Parameters; [Kit] Component Data; Stock, Resupply & Options [(Kit and Model]; Flying Hour Scenario [Kit and Model]) and the Performance Report Window. Appendix B — Data Dictionary — presents definitions, other field information, and file location information for each of the abbreviated field names.

The actual equations and assumptions underlying the spares optimization computations for ISAAC are described in a separate document called

CHAPTER 2

Basic Requirements Model Runs

INTRODUCTION

The purpose of ISAAC is to develop the most cost-effective initial provisioning spares mix (shopping list) that meets a set performance target (aircraft availability) or constrained budget target for a given weapon system. That spares mix is to support all the procured aircraft for a user-specified period including non-war and war conditions. This chapter describes the information needed to develop the spares mix. The model is designed so that you can easily develop the mix by setting only a handful of parameters, which we discuss in this chapter. Chapter 3 will describe more advanced parameters that the user will change only for special analyses or when first preparing the input (a kit) for a particular weapon system.

Specifically, this chapter presents the following basic information:

- The system hardware and software requirements for ISAAC.
- A description of basic model operation and key model input parameters.
- A summary of model output reports.

SYSTEM REQUIREMENTS

ISAAC requires the following personal computer (PC) hardware and software (not included with the ISAAC installation disks):

- A 486DX-based computer certified for use with Microsoft® Windows Version 3.x. The model can also use a 386- or 486SX-based computer with an Intel 80 x 87 math coprocessor, but 386 machines will run more slowly.
- An EGA, VGA, or super VGA graphics card compatible with Microsoft Windows Version 3.x.
- A Microsoft Windows-compatible mouse.
- A minimum of 8 megabytes (MB) of random access memory (RAM).
- A minimum of 540 kilobytes (KB) of free conventional RAM.

– A minimum of 25 MB of free disk space.

– Microsoft Windows Version 3.x or higher and Microsoft MS-DOS® Version 3.3 or higher installed on your computer. The MS Windows options should be set to a 32-bit disk access and a windows swap file of at least 20 MB. You can see and set those options by following the sequence of steps set forth below; however, before starting this sequence, you should close all Windows applications, because this sequence will restart Windows:

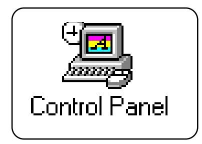
► From the Windows Program Manager select **Control Panel** (accessed via icon).

► Select **386 Enhanced** icon.

► Click on the **Virtual Memory** button.

► Click on the **Change>>** button.

► The "Virtual Memory" Window will then appear. Click on the **Use 32 Bit Disk Access** check box and enter an "**x**".



► Also from the same "Virtual Memory" Window, enter a swap file size of at least 20,000 KB and click on the **OK** button.

► Another "Virtual Memory" dialog box will then appear. Click on the **Restart Windows** button.

– In the config.sys file, set "Files = 90," allowing 90 files to be open simultaneously.

If your machine has the S3 Video Card, when you start ISAAC, a general default error message will appear, or else ISAAC screens will not appear correctly on your screen. If this is the case, you need to change your monitor display setting to the standard **VGA,** using the following sequence:

– Set Up a screen display of 640x480x16 and ensure the following settings in your Windows Setup. Before starting this sequence, you should close all Windows applications, because this sequence will restart Windows.

► From the Windows Program Manager select **Windows Setup** (accessed via icon).

► Choose **Options** from the pull-down menu.

- Select **Change System Settings . . .**.
- Select ... **640x480 sm. fonts** from the **Display** drop-down list box.
- Click on the **OK** button.
- The “Change System Settings” Window will appear. Click on the **Current** button.
- The “Exit Windows Setup” Window will then appear. Click on the **Restart Windows** button.

Model Operation

When you enter ISAAC, you see two choices highlighted, on the left side of the main menu: **Kit** and **Model**. The **Kit** menu lets you develop baseline model inputs from other sources. These inputs could be data imported from other databases or generated by other models such as the IPSS (Initial Provisioning Support System), which we will describe in Chapter 4: Developing Input Kit Data. Clicking on this menu item opens the Kit Parameters Screen, enabling you to import a new kit or modify an existing one. The **Model** menu option allows you to run ISAAC either to estimate spares requirements or to evaluate a spares kit in terms of its impact on availability. This chapter will focus entirely on estimating spares requirements. Chapter 8 will describe the evaluation of spares kits.

The model generates spares requirements using one of two potential sources of information. The basic source is a list of imported baseline kits developed via the kit option. The second is a library of previous runs, as we describe next. We will first treat the entire process of running the model from the previous run. Then we will discuss all the basic input and output, how to view the input and output, and how to store and retrieve previous work.

Basic Requirements Run Sequence

- Choose **Model** (from the menu bar). The model automatically selects the latest run from the library of previous runs.
- Click on the **Modify** button (on the bottom left of the Model Parameters Screen).
- Enter a **Run Description** (a new run description is required to run the model).
- Change 1st Analysis Day & 2nd Analysis Day information on the Model Parameters Screen as necessary.

- Click on the **Scenario** button (on the right of the Model Parameters Screen). Close the Flying Hour Scenario Screen by clicking on the **Close** bar when finished.

- Click on the **Run Requirements** button. The ASM Window will appear within a few seconds. Once the ASM processing has been completed, the Processing Model Output Window will be displayed.

- The Spares Cost Summary a PLTT Before Spares Delivery Screen will then appear. For now we will let the model generate the annual cost estimates so just press the **F10** function key. Then close the screen by clicking on the Control-menu box. The Processing Model Output Window will reappear while the model completes its output processing.

- Performance Report appears (click on the close button in the middle of the window).

Model Input

ISAAC requires various inputs. The largest part is the component input, which is generated in the **Kit** menu and cannot be altered while in the model mode. However, you may change other input that we term model parameter information.

PARAMETERS

The Model Parameters Screen (Figure 2-1) contains all the system (or global) parameters for a particular ISAAC run; these may include weapon system name, the days to be analyzed, the weapon system availability target (expressed as NMCS), the budget constraints, repair and resupply assumptions, and the flying hour scenario.

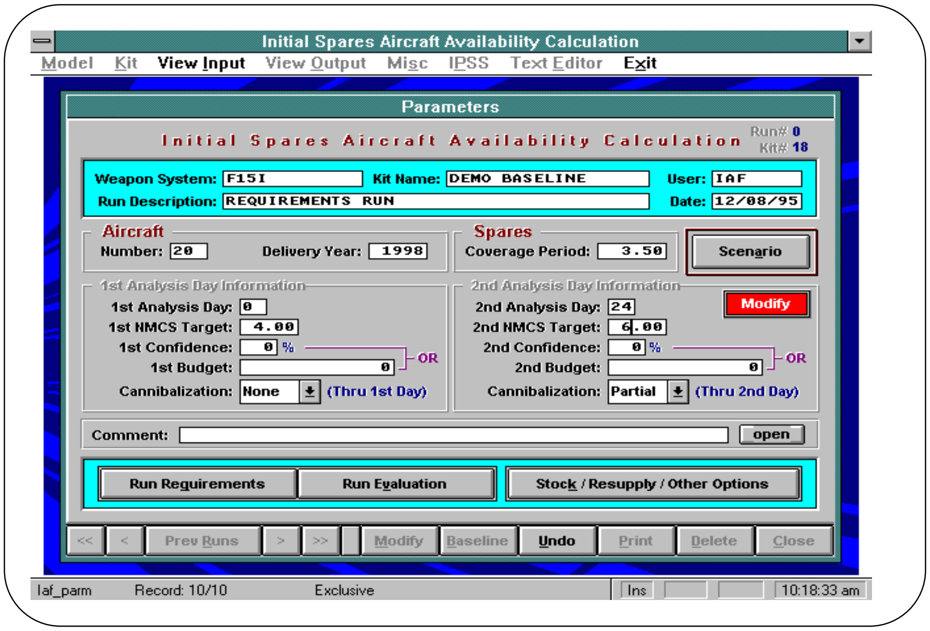
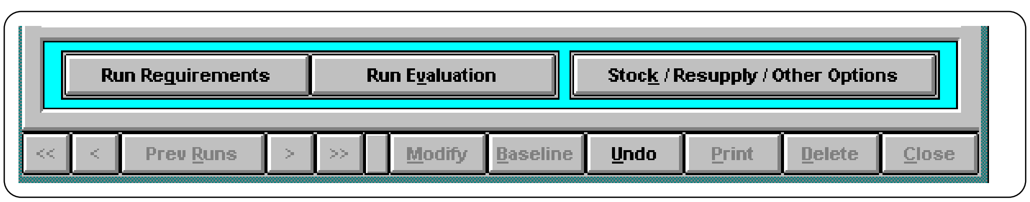


Figure 2-1
Model Parameters Screen

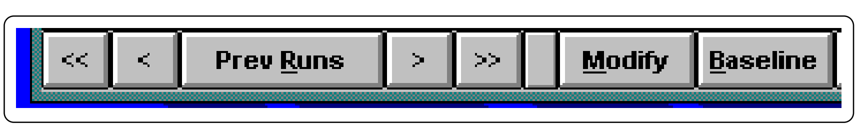
We will divide the screen into three sections (excluding the menu bar) for ease of description. We will introduce the sections in the order in which they must be used to run the model.

Lower Box

The first step in running ISAAC is to select the input that you want to run through the model. This selection is made in the lower box of the Model Parameters Screen (depicted below).



You must start by selecting one of the three labeled buttons located at the lower left of your screen.



Prev Runs — Selecting this button opens the library of all the previous ISAAC model runs that have been run on your PC. This screen is the Find Model Run

Window (Figure 2-2). Follow the instructions at the top of the screen to select a previous run as the starting point for your run.

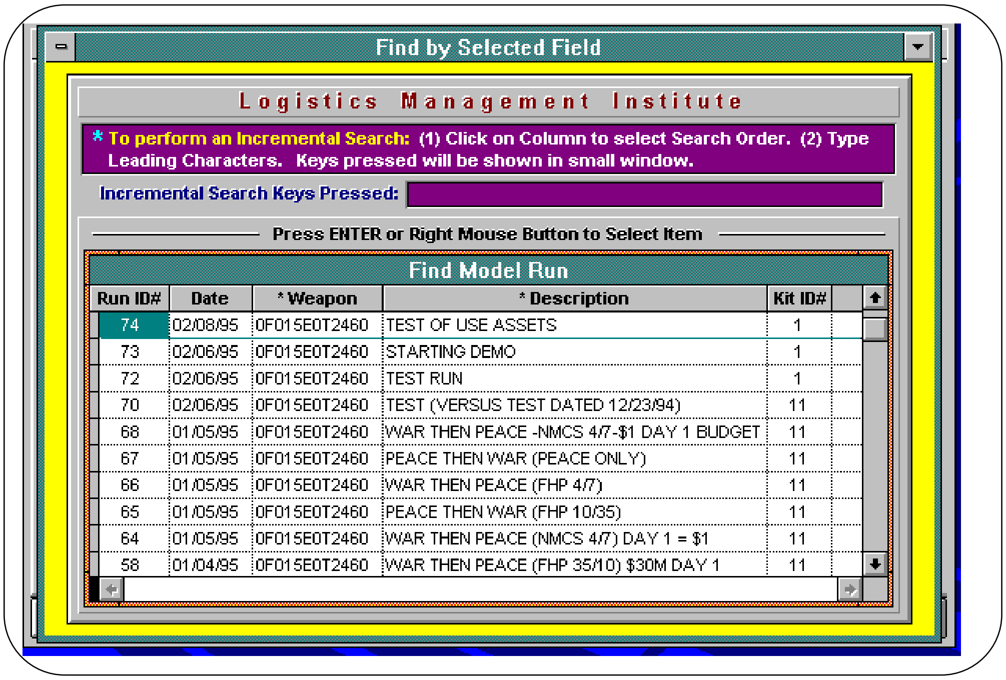


Figure 2-2
Find Model Run

Alternatively, you can use the two unlabeled buttons on either side of the **Prev Runs** (see Figure 2-1) button to move around the library of previous model runs without viewing the runs through the Find Model Run Window. The advantage to using these buttons is that doing so allows you to view the run time parameters for the previous runs in the familiar Model Parameters Screen format. The unlabeled buttons operate as follows:

– Selecting the > button will fill the Model Parameters Screen with the run time parameters from the previous model run immediately above the currently displayed model run in the Run Library.

– Selecting the >> button will fill the Model Parameters Screen with the run time parameters from the run at the top of the Run Library.

– Selecting the < button will fill the Model Parameters Screen with the run time parameters from the previous model run immediately below the currently displayed model run in the Run Library.

– Selecting the << button will fill the Model Parameters Screen with the run time parameters from the run at the bottom of the Run Library.

Modify — Selecting this button will require you to enter a unique run description (see **Delete** later in this section). (ISAAC will not allow two or more runs to have the same run description. Therefore you are required to select **Modify** even if you do not want to vary any of the input parameters.) Once you enter a new run description, you may change the values of any of the fields in the middle box of the Model Parameters Screen, the user name in the upper box, and the scenario information and fields accessed via the **Stock / Resupply / Other Options** button (discussed in Chapter 3).

Baseline — Selecting this button will enable you to select a new baseline of input parameters. A baseline consists of a kit and all associated input parameters, including defaults for the Model Parameters Screen. The baseline selection allows the user to start from or return to a set starting point. It also allows the user to store different kits (e.g., F-15I, F-16C . . .) and easily access them when needed. This window is the Available Kits Window (Figure 2-3). Follow the instructions at the bottom of the window to select a new baseline kit as the starting point for your run.

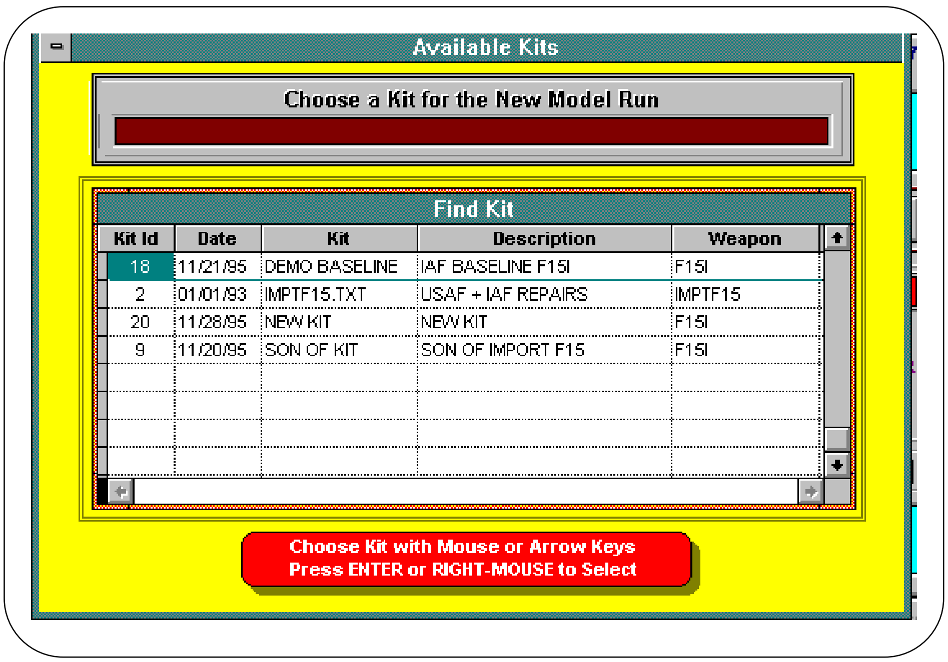
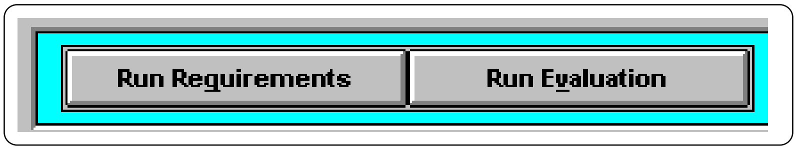


Figure 2-3
Available Kits Window

The next group of buttons are used to start model processing.

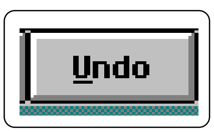
Run Requirements — Starts the model processing to determine the requirements in terms of dollars and spares necessary to meet the optimization target, expressed as ENMCS or confidence level or budget constraint.



Run Evaluation — The requirements capability of ISAAC enables the user to enter a target (cost or availability) so that ISAAC can calculate the spares mix required to meet that target. ISAAC's evaluation capability approaches the spares mix problem from the opposite direction — ISAAC enables the user to select a spares mix so that ISAAC can calculate the availability that mix provides. The evaluation mode is usually used to determine how well a spares mix performs under conditions that are different from those of the original requirements run, such as a different flying hour scenario(see Chapter 8: Evaluating Spares Mixes with ISAAC).

Stock / Resupply / Other Options — These options are the more advanced parameters for running ISAAC; the user will change them only for special analyses or when first preparing a weapon system kit for spares analyses (see Chapter 3: Advanced Requirements Model Runs).

Undo — Deletes the previous parameter changes and brings back the original parameters of the selected run.



Print — Prints all the parameters on the Parameters Screen (Figure 2-1), Flying Hour Scenario Screen (Figure 2-6), and the Pipeline, Resupply, & Options Screen (see Chapter 5 for more instructions on how to print model run input and output data).

Delete — Deletes all records associated with this particular **Run Description,** including all output information from the run and the global input information (input via the Model Parameters, Model Flying Hour Scenario, and Model Pipeline, Resupply, & Options Screens and maintained in the **Run Log**) If you want to delete many runs, we suggest using **Delete** in conjunction with the buttons to the right and left of the **Prev Runs** button. Note, that the item-level input information cannot be deleted by using this **Delete** button but only through the Kit menus.

<u>C</u>lose — Closes this screen and returns you to the initial screen.

UPPER BOX

After selecting the item-level input that you want to run through the model, you next describe your run in a unique manner in the upper box (Figure 2-4).

Figure 2-4
Upper Section of Model Parameters Screen

Weapon System: — This is the name you entered when creating a kit; it is also found in the Nhansn (next higher assembly national stock number) field for each LRU record. It is typically an identifier for the aircraft or weapon system. The user cannot change this field.

Run Description: — This must be a unique description of the run (no two runs may have the same description). Long narrative descriptions may be entered in the **Comment** field in the middle box.

Kit Name: — This is a name associated with a particular kit. It is entered by the user during the kit creating process. The user cannot change this field.

User: — User ID will be entered by the user.

Date: — Will be updated by the model. The user cannot change this field.

Run # — Run # is a unique number assigned to each model run for tracking and identification purposes in the library of previous runs. This field will be **0** when you bring in a new baseline kit (see Chapter 4). Otherwise it will contain the model run number corresponding to the model run currently open. If you click on the **<u>M</u>odify** button, the field will contain the model run number of the run you are modifying. The user cannot change this field.

Kit # — The kit # is a unique number assigned to each baseline kit for tracking and identification purposes in the library of available kits. The user cannot change this field.

MIDDLE BOX

The middle box (Figure 2-5) is used for entering the processing parameters for the model run.

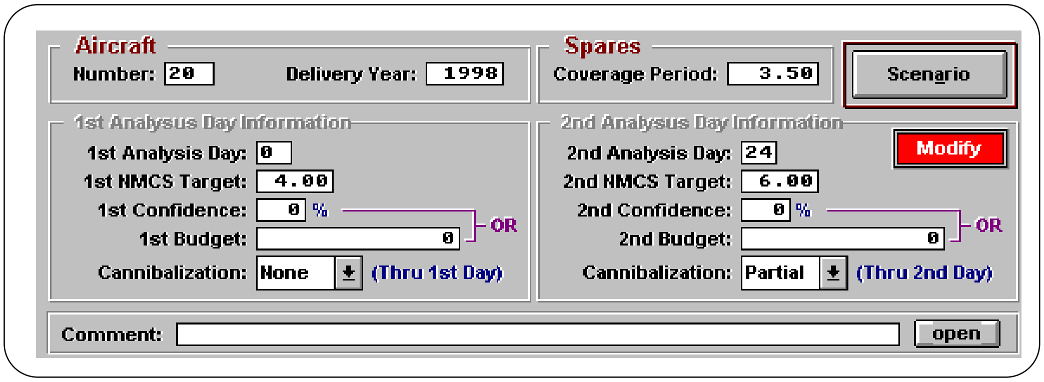


Figure 2-5
Primary Global Parameters

The middle box contains the following primary global parameter fields:

– Aircraft

► **Number:** — The quantity of aircraft at each base (PAA — Primary Authorized Aircraft). Must be an integer from 1 through 999.

► **Delivery Year:** — The calendar year in which the aircraft will be delivered in. The model uses the first day of this year as the end of the budget period and the beginning of the spares coverage period.

– Spares **- Coverage Period:** — This is the time during which the initial provisioning system is responsible for procuring spares for this weapon system. The sole impact of increasing the coverage period is to increase the forecasted number of condemnations we consider and thus the number of spares required to replace those condemnations. Using only the expected condemnations in an item’s PLTT (the basic assumption used in replenishment models) underestimates the condemnations if the coverage period is longer than the PLTT and overestimates them if it is shorter. Thus, ISAAC uses an additional accounting factor called CondAssets where the sum of the CondAssets and the condemnation pipeline equals the expected number of condemnations in a coverage period. Besides that adjustment, ISAAC must also adjust for the fact that the number of daily flying hours over the coverage period is not constant. Figure 1-6 plots a nominal example of flying hours for a non-war scenario over the coverage period. In this example, the only year during which flying hours change is the delivery year; that is because not all the aircraft are delivered at one time. We make a key simplifying assumption that aircraft delivery is spread out evenly over one year.

Thus, the first year has one-half the flying hours of each of the successive years, which means that the actual number of years of steady-state flying is the coverage period minus half a year. Therefore, ISAAC automatically subtracts 0.5 years from the coverage period entered by the user. (If the aircraft are delivered over more than one year or all in one delivery, you must calculate the number of years of steady-state flying hours in the coverage period *and* add 0.5.) The user must also consider three additional factors when entering a coverage period:

- The user must set the **Include Starting Assets?** drop-down list box to **Use Assets** in the advanced parameters (Chapter 3) to procure spares for the entire coverage period. If you set the **Include Starting Assets?** drop-down list box to **- Do Not Use Starting Assets**, the model will not allow you to enter a coverage period greater than 0 until the **Use Assets** option is selected.

- The user cannot enter a coverage period value between 0 and 1 year, because that time frame includes dynamic flying hours (see Figure 1-6) and thus is inconsistent with the basic assumption of steady-state flying hours that the model makes for the non-wartime period.

- If the user enters a “0”, the model treats that value not as a time period but as a switch to use a different logic in calculating condemnations. A zero causes ISAAC to ignore the coverage period and its adjustments and estimate condemnations solely on the basis of each item’s PLTT.

The ISAAC model is designed to estimate spares requirements to support both wartime and non-wartime scenarios. This process is initiated by choosing two analysis days (see Figure 2-5 and the Model Approach subsection of Chapter 1) each representing a cumulative period up to and including that day. A “0” for the model’s analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A “24” translates into having the model procure spares for a 24-day war scenario that starts at the end of the coverage period and ends 24 days later (alternatively, any other day of the wartime scenario could be entered). While the basic operation generally uses a day of non-war and the last day of the war, the model is flexible and can operate on any one or two days of interest.

– 1st Analysis Day information

- **1st Analysis Day: —** (REQUIRED INPUT) The first day to be analyzed. Must be an integer from 0 through 99. An entry of 0 will use the non-war, item-level input data and the non-war flying hour scenario. Any other analysis day (1 through 99) will use the wartime item-level input data and the wartime flying hour profile and optimize through that point.

- **1st NMCS Target: —** (REQUIRED INPUT) The allowable number of not mission capable for supply (NMCS) aircraft on the first analysis day. The model minimizes the spares cost required to meet the NMCS target. The NMCS target serves two functions: it acts as a curve target to stop the optimization process once the target is reached (in a manner similar to that for a budget target, described below) and it identifies the number of aircraft that are likely candidates for cannibalization according to the maintenance assumption (see below). The NMCS number must be a positive real number equal to or less than the number of aircraft.

- **1st Confidence: (%) —** This is the probability, expressed as a percentage, of meeting the NMCS target (previously entered). For instance, if your NMCS target is 3, then it is the probability of having 3 or fewer planes down for lack of spares. The model optimizes the probability that the number NMCS is not greater than the target. This is an optional input and is mutually exclusive of the **1st Budget** constraint (i.e., you may enter a confidence % or a budget target but not both).

- **1st Budget: —** This is the budget constraint that is to be applied to the **1st Analysis Day:**. An optional input, it is mutually exclusive of the **1st Confidence: %** constraint (i.e., you may enter a confidence % or a budget target but not both). In the budget-constrained mode, the model may not meet the NMCS target on either or both days but it will not exceed the budget constraint.

► **Cannibalization (Thru 1st day): — Full** indicates that all LRUs may be cannibalized. **Partial** instructs the model to use the individual cannibalization field in each LRU record ('Y' LRUs are cannibalized, while 'N' LRUs are not). **None** indicates that no LRU may be cannibalized. The model allows cannibalization of the 'Y' coded items and reports these actions as "easy" cannibalizations. A spares mix built using any cannibalization optimization will be radically different from one of equal cost optimized for non-cannibalization. *Note: the model assumes that SRUs are always cannibalized in the repair shops*

– 2nd Analysis Day information is used only when the user wants a two-day analysis (non-war and war scenario, or an analysis of two war days). All variables have the same definitions (see above) as their 1st Analysis Day information counterparts except for the following:

- **2nd Analysis Day: —** If you want only a one-day run, enter a space as input for this field. When you move off that field, the model will insert an "*" to indicate that no second-day analysis is performed.

- **2nd Budget: —** This is the incremental budget constraint that is to be applied to the 2nd Analysis Day. The total combined budget equals the value entered here plus the value entered for the 1st Analysis Day.

- **Comment** — Enter here any descriptive data pertaining to the run.

- **Scenario** button

This button accesses the Flying Hour Scenario Screen (Figure 2-6).

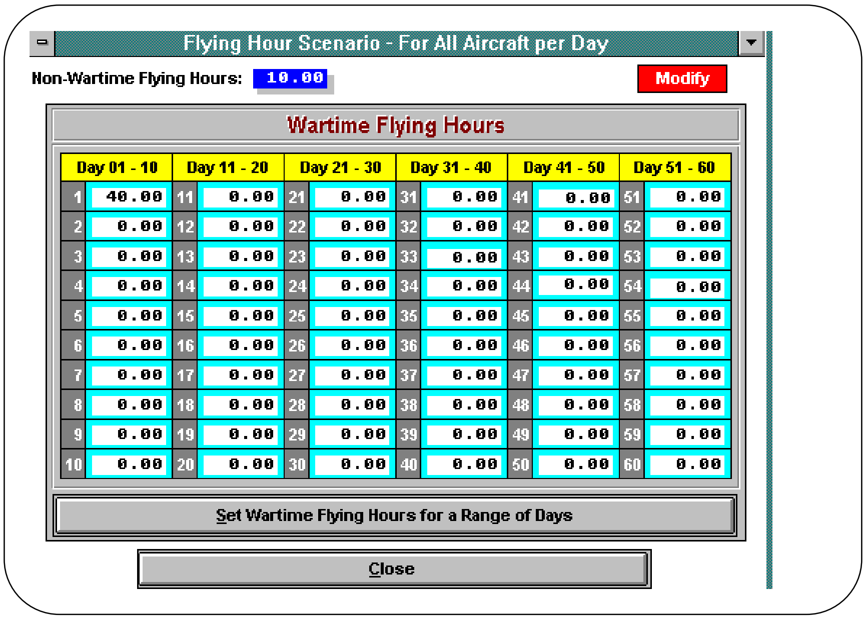


Figure 2-6
Flying Hour Scenario Screen

Flying hours are the combined number of flying hours all the aircraft fly in a day. The model multiplies that number by the demands per flying hour and the quantity of the item per aircraft (item data) to obtain the total demands for an item for each day.

– **Non-Wartime Flying Hours** — The model uses this daily flying hour total to compute requirements for the steady-state non-war period (the coverage period) that precedes the war. This value is required if you run Day 0 as an analysis day or you want steady-state conditions to precede the war scenario.

– **Day 01 – 60** — These fields contain the total flying hours for day 01 through day 60 for wartime conditions. You are required to enter the flying hours for each day through the day after which the flying hours remain constant until the end of the war. If the flying hours remain the same throughout the war, you are required only to enter a value on Day 01. The model is programmed to treat the flying hours as constant from the last non-zero value entered. The model will accept a break in the flying hours (i.e., one or more days with 0 flying hours in between two or more days with flying hours greater than 0).

– **Set Wartime Flying Hours for a Range of Days** bar — This bar is used to set the flying hours as a constant over a range of days in wartime. When you click on it, the dialog box shown in Figure 2-7 appears.

Figure 2-7
Flying Hours — Range of Days

Model Output

Summary Output

Once the model has finished running, the first screen you will see is the resulting estimated spares procurement costs by year. Suppose the model is run to determine which spares are required given a total budget constraint of $100 million. The model then estimates when each spare needs to be ordered. To do that, the model starts by assuming that the manufacturer delivers all spares when the first aircraft are delivered (for simplicity, we assume at the beginning of a year — for our example, 1998). Then, by moving an item's PLTT back in time (the PLTT is given data), the model estimates the year in which the order needs to be placed (see Figure 2-8).

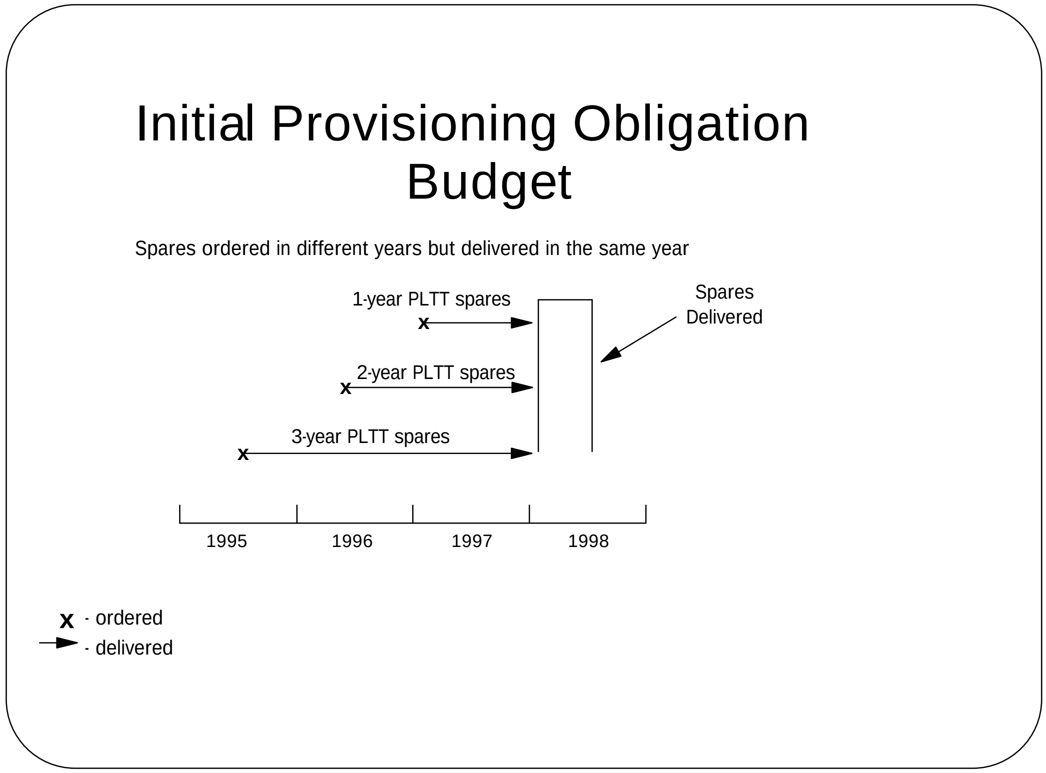


Figure 2-8
Initial Provisioning Obligation Budget

The model also estimates how the total spares buy cost is best obligated by year. For instance, all spares with less than a 1-year PLTT are ordered in 1997 and added to the 1997 spares buy cost, all spares with a 1-to-2 year PLTT are ordered in 1996 and added to the 1996 spares buy cost, and so on.

An additional consideration is that the IAF budget constraint can have two parts: a total budget constraint (e.g., $100 million) for all initial provisioning spares, and an annual obligation constraint for each year (e.g., $20 million, $30 million, and $50 million for 1995, 1996, and 1997, respectively). Once the model adds up spares buy cost obligations by year, the sum may be greater than the total budget constraint for one or more years. For instance, the model might generate annual buy cost of $20 million, $50 million, and $30 million, while the budget constraints are $20 million, $30 million, and $50 million dollars for 1995, 1996, and 1997, respectively. To match the annual budget constraint, the model must move $20 million of spares from 1996 back to 1997 or delay the delivery of some of the 2-year PLTT spares by 1 year (see Figure 2-9).

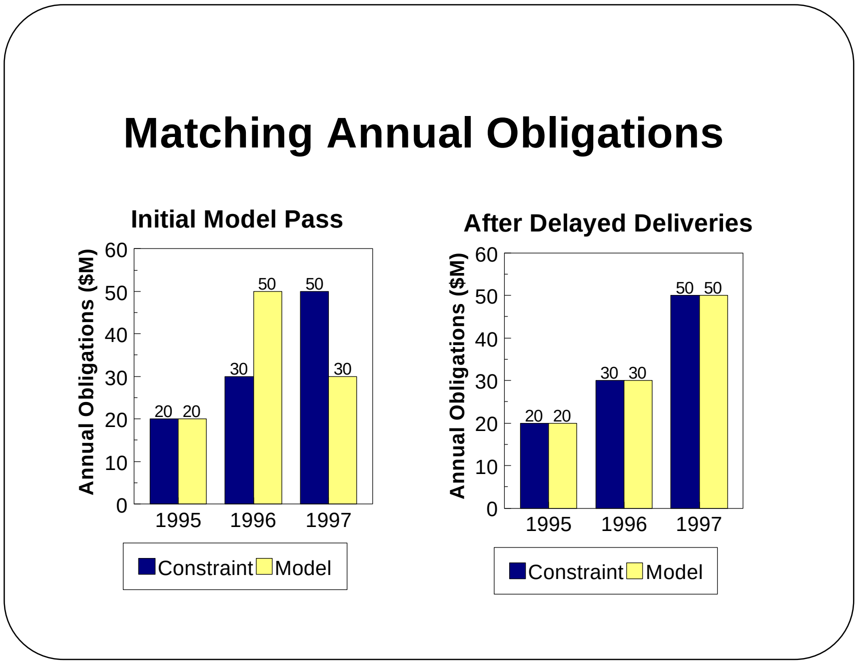


Figure 2-9
Matching Annual Obligations

The reason why the model can delay the delivery of spares is that it starts by assuming that the manufacturer delivers all spares simultaneously with the first aircraft. Yet the spares requirement is for the entire coverage period. Thus, for items with possible condemnations, the model requirement is for the entire coverage period (e.g., 3.5 years), but the item in the first year needs only part of that requirement. Therefore, initially, the model projects delivery of some spares before they are needed; those are the spares the model delays to match the budget target. For instance, if a consumable item's PLT is 2 years, part of the item's requirement is for the first year (1998) and part is for the second year (1999). The spares needed in the second year are the candidates for delayed delivery (delayed until 1999).

The Spares Cost Summary a PLTT Before Spares Delivery Screen (Figure 2-10) is the mechanism you use to enter annual budgets and have ISAAC delay spares deliveries to match those budgets. We now describe the steps required to implement our hypothetical example in Figure 2-9. In our example, the model output of $20 million, $50 million and $30 million occurs in the column labeled **Buy Cost** in Figure 2-10. You enter your budgets of $20 million, $30 million and $50 million in the **Budget** column and presses **F10.** The model then tries to match those budgets by delaying spares deliveries; produces new buy costs approximately equal to $20 million, $30 million and $50 million and displays them in the **Buy Adjusted** column (Figure 2-10). The **Delta Adjusted** is the actual adjustment for that year (in our example, it equals $20 million for 1996).

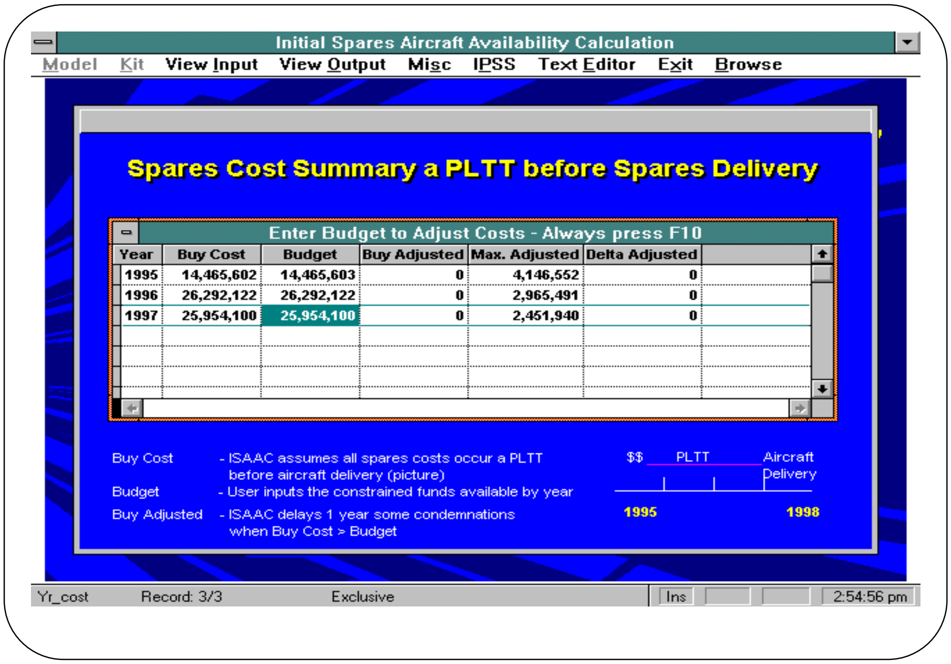


Figure 2-10
Spares Cost Summary a PLTT before Spares Delivery Screen

In Figure 2-10, the **Max. Adjusted** column shows the greatest number of dollars that can be adjusted in each respective year. The amount displayed represents purchases of spares to replace condemnations that are expected to occur during the coverage period of the model run. The dollars displayed indicate those purchases that can be delayed, without adversely affecting availability. If the cost estimate minus the budget is greater than the Max. Adjusted figure, the model cannot meet the target. In that case, other model runs with smaller coverage periods are required to determine how to meet that estimate. Notice, too, that the model does not order spares earlier than needed if the budget exceeds the model buy cost estimate. The assumption is that for large uncertainties in initial provisioning, it is always better to delay decisions and budget expenditures, if possible.

The output presented to the user at the end of a model run is the Performance Report (Figure 2-11), which contains summary information on the last model run. If accessed from the Model Parameters Screen, it summarizes information pertaining to the selected model run description as displayed on the parameters screen.

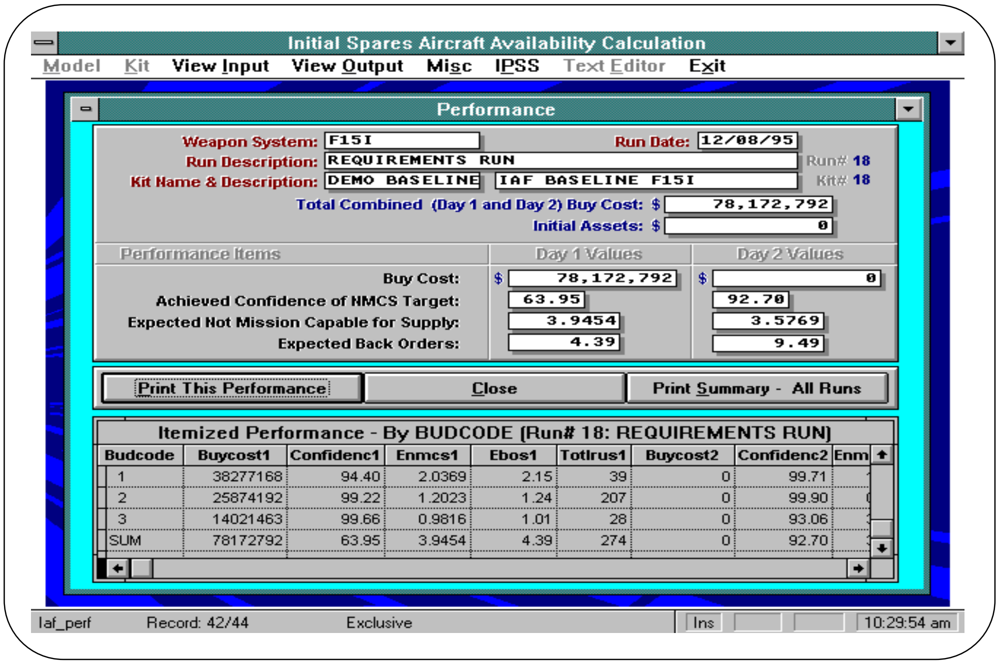


Figure 2-11
Performance Report Window

The following information is included in the top of the Performance Report:

– **Weapon System:** — from the Model Parameters Screen for this model run.

– **Run Description:** — from the Model Parameters Screen for this model run.

– **Run Date:** — from the Model Parameters Screen for this model run.

– **Kit Name & Description:** — from the baseline kit information.

– **Total Combined (Day 1 and Day 2) Buy Cost: ($)** — this equals the sum of the buy costs for the two analysis days.

– **Initial Assets: ($)** — existing assets; that is, the dollar valueof all item assets specified in the InitAsset field for each item. The InitAsset field can be viewed in the Component Data browse window and is described in Appendix B.

The following information is included in the middle of the Performance Report, in the block labeled “Performance Items” (for analysis Day 1 and Day 2 Values)

– **Buy Cost: ($) —** the purchase cost of the required spares, broken out by analysis day.

– **Achieved Confidence of NMCS Target: —** the confidence level of meeting the target NMCS value for each analysis day; given the buy cost plus initial assets associated with that day.

– **Expected Not Mission Capable for Supply: —** the expected (average) number of NMCS aircraft for each analysis day, given the buy cost plus initial assets associated with that day.

– **Expected Backorders: —** the forecasted performance value for each analysis day, given the buy cost plus initial assets associated with that day.

The following information is included in the bottom of the Performance Report, in the block labeled “Itemized Performance — By BUDCODE” — This table breaks out key outputs for each analysis day by Budcode.

► **Budcode —** The Budcode is a user-defined budget code from 1 through 99 used for recording composite output parameters for a group of items. The budcode typically defines a particular subsystem (e.g., fire control or landing gear) or other group of LRUs (e.g., the LRUs for a particular item manager).

  – Each budcode represented in the item data will be listed in this section of the Performance Report. The values presented in the columns to the right of each Budcode (e.g.,**Buycost1**, **Confidenc1,** etc.) will be composite values for all items making up the respective Budcode (the model assumes that all SRUs have the same Budcode as their parent LRUs).

  – The last Budcode in the table is the SUM Budcode, which contains the cumulative results of all Budcodes (i.e., the results for the aircraft under consideration). This record contains the total buy costs, EBOs, ENMCS, and number of LRUs. Each value is a sum of the individual Budcode values except forthe confidence level SUM, which is the product of the confidence level values.

► **Buycost1** and **Buycost2 —** the buy cost of the spares required to meet the availability target on the 1st and 2nd day to be analyzed, respectively.

► **Confidenc1** and **Confidenc2 —** this is the percent confidence of meeting the NMCS day 1 or day 2 target given the total buy cost plus initial assets for all spares with the same BUDCODE (subsystem). For SUM, this is the confidence level of meeting the target NMCS value for the entire aircraft, given the buy cost plus initial assets associated with both days and for all spares regardless of BUDCODE. (Any analysis day with a no-cannibalization policy will yield a non-intuitive confidence level value. This is because the confidence level calculation is not as well behaved a function in the no-cannibalization case as in the cannibalization case.)

- **Enmcs1 and Enmcs2** — The number of systems expected to be not mission capable, because of a shortage of spares on the day to be analyzed.
- **Ebos1** and **Ebos2**— The number of expected backorders on the respective day analyzed.
- **Totlrus1** and **Totlrus2**— The total number of LRUs in the system. If an LRU has an excess number of spares, the model may ignore it in the calculation to speed processing time and not include it in this table.

Viewing Other Isaac Reports

ISAAC produces a number of reports accessible via the pull-down menus (see Chapter 5: Input, Output, and Miscellaneous Data Windows for more detailed descriptions)..

View Input — Clicking on this menu item opens a pull-down menu with the following menu items:

- **Component Data** — Item-level input data will provide access to the component-specific input associated with the run description displayed on the Model Parameters Screen. Runs using the same kit will view the same component data
- **Run Log** — Model run time parameters for all previous ISAAC runs.

View Output — Clicking on this menu item opens a pull-down menu with the following menu items. Selection of any of these menu items will provide access to output information pertaining to the model run description displayed on the Model Parameters Screen.

- **Pipeline Data** — Pipeline data translates failure rates and repair times into the average number of broken items at various maintenance levels (i.e., the mean base pipeline, the mean depot pipeline, etc.) The pipeline means are given for both analysis days and are one of the most important pieces of information, since they are directly correlated with the total items spares requirement or target.

- **Shopping List Data** — Shopping list information gives the item-by-item spares requirement (target) and procurement (Total buy) needed to meet the NMCS or budget constraint.

- **Performance Report** — Basic model output (covered earlier in this chapter under Output).

- **Curve** — Curve presents the performance measures over a range of costs for the last analysis day .

- **Yearly Cost** — Yearly Cost presents the total buy cost broken out by order year, as described earlier.

Misc — Clicking on this menu item opens a pull-down menu with the following menu items. Selection of any of these menu items will provide access to special reports (that do not fall into the basic categories of input or output) pertaining to the model run description displayed on the Model Parameters Screen (except for the **View Shop Comparison**, which will default to the most recent model run).

- **View Stats** — Clicking on this menu item opens a table of statistics that apply to the model run description displayed on the Model Parameters Screen. The report is for all items in the kit and includes statistics on the average repair times, pipeline means, and spares targets.

- **View Input-Output** — Clicking on this menu item displays fields from the component, pipeline, and shopping list reports by NSN.

- **View Shop Comparison** — Clicking on this menu item compares shopping lists from two previous runs.

Exit — Clicking on **Exit** terminates the ISAAC program.

CHAPTER 3

Advanced Requirements Model Runs

This chapter tells you how to estimate spares requirements using some of the more advanced parameter options. Probably you will change these parameters only for special analyses or when first preparing a weapon system kit for spares analyses. The parameters fall into three categories:

– The first allows the user to alter the model's optimization routine and incorporate specific stock objectives. You can use these parameter settings to force the model to include previous procurements, previously ordered spares, or specific item-managers target levels.

– The second involves suspending of repair or distribution activities at the beginning of the war scenario. These parameters allow the model to reflect wartime assumptions more closely, such as the suspension of spares shipments to bases because, initially, trucks must carry out higher priority missions.

– The third is the miscellaneous or remaining parameters that have an impact on the specific aspects of model optimization. These parameters let you change such model aspects as the number of bases or the distributions used to estimate demands.

Some of the advanced parameters have an impact on all items in the kit, while others affect only those with specific item information. This chapter will focus on how the advanced parameters affect requirements model runs, while Chapter 8 will focus on how some of them affect evaluation model runs.

All of the advanced parameters can be modified only from within the Parameters Screen (accessed via the Model or Kit menu). These advanced options are grouped into the Pipeline, Resupply, & Options Screen, with three sections — Stock Options, Resupply Options, and Other Options. They can be accessed by clicking on the **Stock / Resupply / Other Options** button located in the lower right section of the Parameters Screen.

The Pipeline, Resupply, & Options Screen (Figure 3-1) functions identically whether accessed via the Model or Kit Parameters Screen. The only difference between the kit and model are the steps required to access the Pipeline, Resupply, & Options Screen. This chapter will focus on accessing that screen from the Model Parameters Screen while Chapter 4 will cover how to access the it from the Kit Parameters Screen. The screen can be viewed at any time from the Parameters Screen. This screen can be modified only after the **Modify** button is pressed (as was the case with the basic parameters). The colored box in the upper right of this screen will indicate whether can may only view the fields on this screen or whether you can modify them. The initial values for each field will be the defaults you select when you build the kit (see Chapter 4) or, if you are modifying a previous run, they will be those of that previous run.

In most instances, key team members will modify the advanced parameters associated with each item input database ("kit") when it is first imported; then they will probably not modify those options unless special analyses are required.

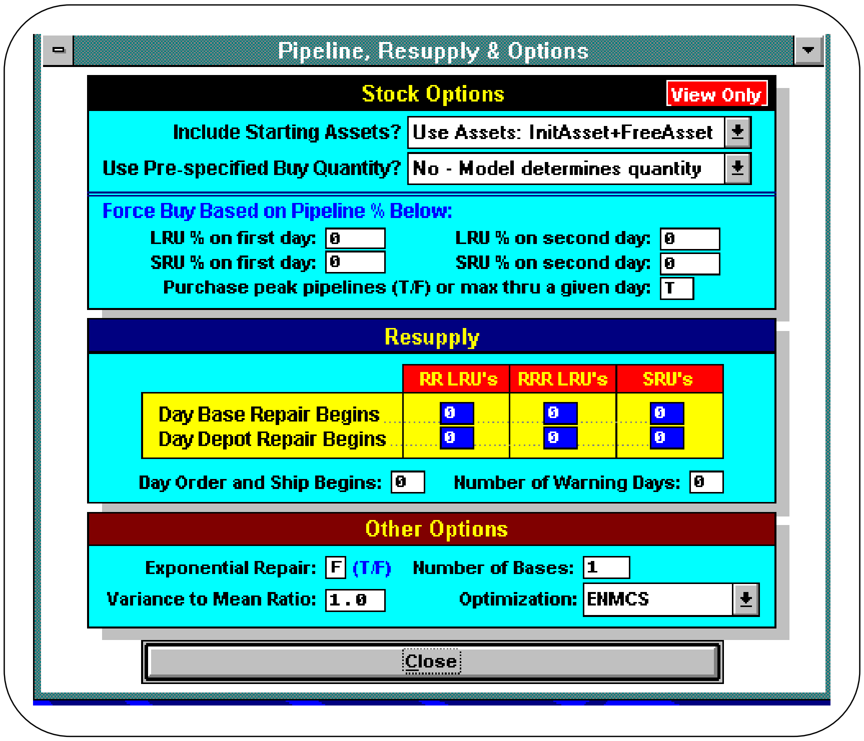


Figure 3-1.
Pipeline, Resupply, & Options Screen

Stock Options

Often the user has previously made stockage decisions on some or all items and wants the model to include those decisions in its solution. The stock decisions break down into two categories: (1) items previously ordered and procured and (2) items ordered previously but not yet procured. We refer tothe former as starting assets, since they do not increase the model buy cost estimate, and to the latter as pre-specified buy quantities (or negotiated, "NEGLV", levels), which force the model to increase the spares cost.

For instance, the IAF has starting assets, previously paid for, in the case of items that are common to the new aircraft and existing aircraft. For pre-specified buys, the item might have outstanding orders already placed because it was an SAIP (Spares Acquisition Integrated with Procurement) item. The stock parameters allow you to include those non-optimal decisions (non-optimal in the sense that the spares quantities may not have been determined on the basis of their contribution to aircraft availability). Furthermore, you can specify starting assets or pre-specified buys as a minimum value and let the model buy more if necessary. Alternatively, you can specify them as a fixed value, and the model will stop purchasing once it reaches the specified level. For pre-specified buy quantities, the model can handle some special cases, such as excluding the item from having an impact on aircraft availability or forcing the model to buy a percentage of the item's pipeline.

Stock Options (Figure 3-2) are accessible via the top section of the Pipeline, Resupply, & Options Screen. The Stock Options section enables you to (1) include or exclude on-hand assets in the model calculations, (2) use predetermined spares target quantities for pre-specified components ; and (3) adjust run time parameters that affect the percentage of the LRU and SRU pipelines that will be bought sacrosanct (i.e., automatically) by the model.

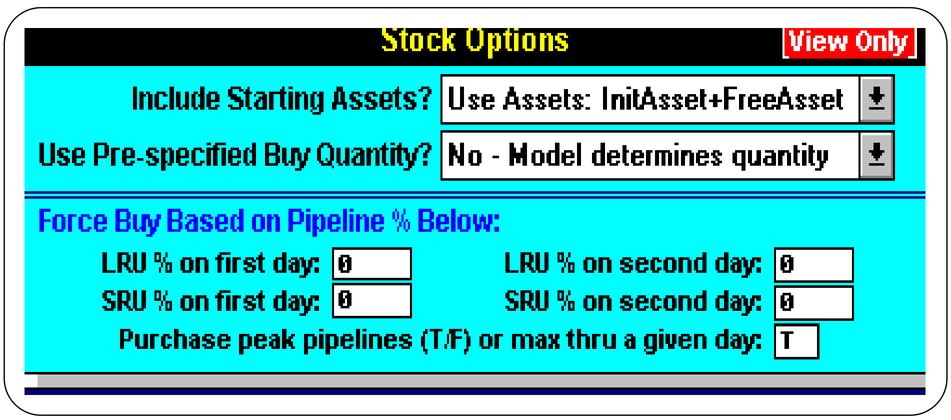


Figure 3-2.
Stock Options (Top Section of the Pipeline, Resupply, & Options Screen)

The initial values for each field will be the defaults you select when you build the kit (see Chapter 4) or, if you are modifying a previous run, they will be those of that previous run.

We will first define each Stock Option field individually. The Stock Option Field Relations section will also provide information on the relationship between the Stock Option fields, item-input data fields, and availability and budget outputs.

Include Starting Assets?

This parameter specifies whether starting stock is considered in the computations. The two options for this field are listed and described below.

Do NOT Use Starting Assets — Use zero starting stock.

Use Assets: InitAsset +FreeAsset — Use starting stock developed from the sum of three separate item-level calculations: (1) the initial assets (what we term *the InitAsset* field in the kit), (2) a representative portion of any assets common to other aircraft (what we term *FreeAsset* -- see Optimizing Spares Support: The Aircraft Sustainability Model [1]), and (3) the assets required to replace the estimated condemnations that will occur during the coverage period and are not included in the condemnation pipeline*(*what we term CondAsset -- treated as a decrement to the other assets and discussed below). While the model includes an item's InitAsset and FreeAsset in calculating availability, it does not include their corresponding costs in the total buy cost, since they will have been procured beforehand. On the other hand, the model treats CondAsset as negative assets that must be procured and includes their costs in the total buy cost output (Chapter 5 — Shopping List Data — describes that information in more detail). The sum of *InitAsset* plus *FreeAsset* minus *CondAsset* is equal to the total starting assets (what we term ITASSE).

Table 3-1 depicts the relationship between the **Include Starting Assets?** field and the component data fields.

Table 3-1.
Relationship Between "Include Starting Assets?", Component Data Fields, and Model Calculations

Model inputs		Impacts		Model output
Stock option field	Component data input[1]			
Include starting assets?	Starting Assets (ITASSE)	Availability[2]	Budget[3]	Spares quantity determined by:
Yes	Quantity	Yes	No	User/model
No		No	No	Model

[1]Blank cells in the matrix indicate that the respective component data input field does not affect the model output.

[2]Having an impact on availability indicates that the component is included in the availability calculation.

[3]Having an impact on budget indicates that the cost of the component is included in the total cost of the spares package.

In Chapter 1 we introduced the coverage period. In Chapter 2 we expanded our description of it and gave an overview of the how the model uses it. We have just reiterated and expanded Chapter 2's description of condemnations as they relate to the requirements calculations. We will now further describe how condemnations and the coverage period are linked. The main impact of the coverage period is to increase the likelihood of condemnations (i.e., the longer the coverage period, the more condemnations can be expected to occur).

Standard replenishment models use the number of condemnation in a PLTT to determine requirements. If the PLTT and the coverage period are the same, that is the proper estimate. But if the PLTT is 4 years and the coverage period is 3 years, the PLTT overstates the number of condemnations. Since no demands occurred in the first year, the model adjusts the PLTT to 3 years (see bottom of Figure 3-3). (Note that the model still uses the full 4-year PLTT to determine when to order the item while using the adjusted PLTT to estimate the number of condemnations.)

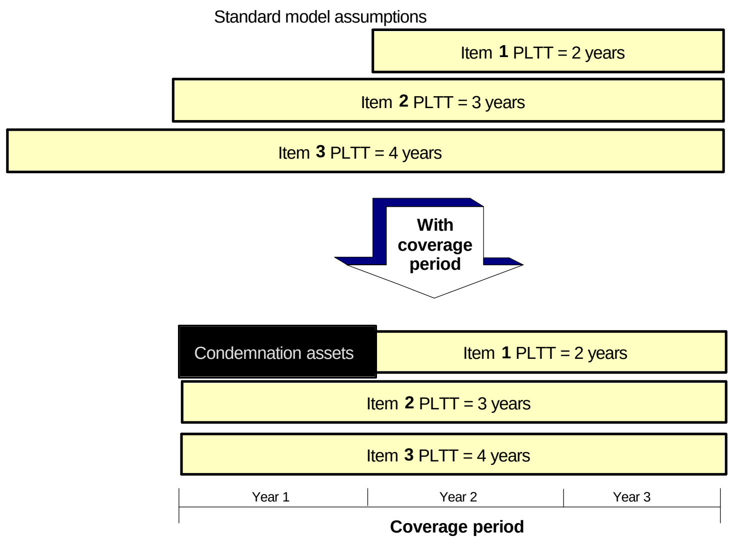
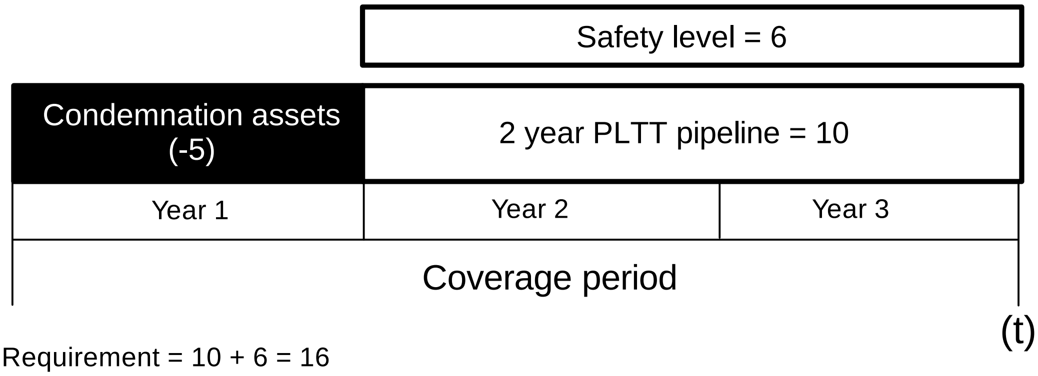


Figure 3-3.
How Coverage Period Adjusts PLTT

When the PLTT is less than the coverage period, if the model were to base the condemnation pipeline solely on PLTT, it would underestimate what will be condemned over the coverage period. To compensate, the model procures additional spares that we term *CondAsset (assets required to replace expected condemnations).* Condemnation assets are part of an accounting tool the model uses to handle the coverage period correctly, as we now describe.

Figure 3-4 displays the model's treatment of condemnations for PLTTs shorter than the coverage period. The model treats these items the same as far as estimating a mean pipeline (average number broken) and the resulting spares requirement is concerned, but it changes the method for calculating the buy quantity. For instance, let us assume that the average cumulative condemnations over the 3-year coverage period total 15. Then, the condemnation pipeline for an item with a 2-year PLTT equals 10 and *CondAssets* equals 5. The model calculates the spares requirement on the basis of the condemnation pipeline and says that it equals 16. However, now the model treats the first year's condemnations as negative assets. Thus, in our example, the buy quantity equals the requirement (16) minus the condemnation assets (-5), or a buy of 21 spares.



Requirement = 10 + 6 = 16
Buy quantity = 16 - (-5) = 21

Figure 3-4
Adjusting the Model When PLTT < Coverage Period

Considering the assets required to replace condemnations — as opposed to simply considering the PLTT in the coverage period calculations — correctly treats the safety level (the difference between the requirement and the pipeline). The model assumes that, for exceptional cases, spares requirements greater than those the model forecasts during the first year may arise. In those cases, the IAF will be able to place additional orders on the supply system against the safety level. Thus, the safety level is needed only during the last 2 years, to protect against unforeseen conditions that preclude additional procurements before Day *t*.

Use Pre-specified Buy Quantity?

This global parameter interacts with two item fields in the kit database: the “NEGLV” field and the “NOP” field. The NEGLV field is the item-specific spares quantity the user wants to include in the spares solution. The NOP field specifies how the NEGLV quantity is treated by the model in terms of spares purchase quantities and inclusion in or exclusion from the availability calculation. The spares quantity may be a minimum, a maximum, or a fixed value (see Table 3 – 2)

No — Model determines quantity — The model will determine the quantity for all items except those with a NOP field value of 'FIX' or 'NOP.' For items with a NOP field value of 'FIX,' the model will use the quantity specified in the NEGLV field as the exact spares target quantity. For items with a NOP field value of 'NOP,' the model will use the quantity specified in the NEGLV field as the spares ceiling; this level is determined by the availability calculation but does not have a direct impact on weapon availability. For instance, the IAF might want ISAAC to determine what levels certain special 'NOP' items will have on the basis of their benefit/cost ratio but might not want to include them in the ENMCS output because they do not have an impact on availability.

Yes — Buy quantity = Item's Neglv (from component data record) — Causes the spares target to be at least the NEGLV quantity (i.e., a floor) for all items with a NOP value *not* equal to 'FIX' or 'NOP' (we usually specify the NOP value in this case as "AAA"). For items with a NOP field value of 'FIX,' the model will use the quantity specified in the NEGLV field as the exact spares target quantity and will use the respective spares in the aircraft availability calculation. For items with a NOP field value of 'NOP,' the model will use the quantity specified in the NEGLV field as the exact spares target quantity but will not use the respective spares in the aircraft availability calculation. For items with a NOP field value of 'ORD,' the model will use the quantity specified in the NEGLV field as the quantity of spares already placed on order (a minimum buy total) and to be delivered in the analysis period.

Table 3-2 depicts the relationship between the **Use Pre-specified Buy Quantity?** field options and the component data input fields.

Table 3-2.
Relationship Between "Use Pre-specified Buy Quantity" Field, Component Input Data, and Spares Output Quantities

Model inputs Parameter stock option field	Kit inputs Component data input fields[1]		Impact on spares solution		Model output
Use Pre-specified Buy Quantity?	NEGLV (user-specified target quantity)	**NOP**	**Availability**[2]	**Budget**[3]	**Component level spares target is the quantity specified in NEGLV?**
No		'AAA' or NOP'	Yes	No	No — Model determines
No	Quantity	'FIX'	Yes	Yes	Yes
No	Quantity	'NOP'	No	Yes	No — NEGLV quantity is a ceiling
Yes	Quantity	'AAA'	Yes	Yes	No — NEGLV quantity is a floor
Yes	Quantity	'FIX'	Yes	Yes	Yes
Yes	Quantity	'NOP'	No	Yes	Yes
Yes	Quantity	'ORD'	Yes	Yes	No — NEGLV quantity is a buy quantity floor

[1]Blank cells in the matrix indicate that the respective component data input field does not affect the model ou

[2] Having an impact on availability indicates that the component is included in the availability calculation.

[3] Having an impact on budget indicates that the cost of the component is included in the total cost of the sp

Table 3-3 summarizes the interaction between the **Use Starting Assets?** field and the **Use Pre-specified Buy Quantity?** field. We also give examples of each of those interactions in Table 3-5.

Table 3-3.
Interaction Between Starting Assets and Pre-specified Buys

Stock option field		Component data input fields[1]			Spares quantity	Impacts
Include Starting Assets?	**Use Pre-specified Buy Quantity?**	**Starting assets (ITASSE)**	**NEGLV (user specified target quantity)**	**NOP**	**Spares target is the quantity specified in NEGLV**	**Availability[2] and / or budget[3]**
Yes	**Yes**	Quantity	Quantity	'AAA'	**No —** but NEGLV is the floor	Availability
Yes	**Yes**	Quantity	Quantity	'ORD'	**No —** but NEGLV is the minimum buy total	Both
Yes	**Yes**	Quantity	Quantity	'FIX'	**Yes —** model will buy up from ITASSE quantity to NEGLV quantity	Both
Yes	**Yes**	Quantity	Quantity	'NOP"	**Yes —** model will buy up from ITASSE quantity to NEGLV quantity	Budget
Yes	**No**	Quantity		'AAA' or 'ORD'	**No —** model determines	Neither
Yes	**No**	Quantity	Quantity	'FIX'	**Yes —** model will buy up from ITASSE quantity to NEGLV quantity	Both
Yes	**No**	Quantity	Quantity	'NOP'	**No —** NEGLV is a ceiling. The model may buy up from ITASSE to NEGLV	Both

[1]Blank cells in the matrix indicate that the respective component data input field does not affect the model out
[2]Impacting availability indicates that the component is included in the availability calculation.
[3]Impacting budget indicates that the cost of the component is included in the total cost of the spares package

Table 3-4 expands the information summarized in Table 3-3 to provide the results (in terms of spares quantities and impact on availability and budget) for each combination of Stock Option field values and component input data field values (ITASSE, NEGLV, and NOP).

Table 3-4.
Summary of How Stock Option Field Values in Conjunction with Component Data Input Field Values Affect Spares Output Quantities

Stock optionfield		Component data input fields[1]			Spares quantity	Impacts
Include Starting Assets?	**Use Pre-specified Buy Quantity?**	**Starting assets (ITASSE)**	**NEGLV (user-specified target quantity)**	**NOP**	**Spares target is the quantity specified in NEGLV**	**Availability[2] and / or budget[3]**
No	**Yes**		Quantity	'AAA'	**No —** but NEGLV is the floor	Availability
"	"		Quantity	'ORD'	**No —** but NEGLV is the floor	Both
"	"		Quantity	'FIX'	**Yes**	Both
"	"		Quantity	'NOP'	**Yes**	Budget
No	**No**			'AAA' or 'ORD'	**No —** model determines	Neither
"	"		Quantity	'FIX'	**Yes**	Both
"	"		Quantity	'NOP'	**No —** NEGLV quantity is the ceiling	Both
Yes	**Yes**	Quantity	Quantity	'AAA'	**No —** but NEGLV is the floor	Availability
"	"	Quantity	Quantity	'ORD'	**No —** but NEGLV is the minimumbuy total	Both
"	"	Quantity	Quantity	'FIX'	**Yes —** model will buy up from ITASSE quantity to NEGLV quantity	Both
"	"	Quantity	Quantity	'NOP'	**Yes —** model will buy up from ITASSE quantity to NEGLV quantity	Availability
Yes	**No**	Quantity		'AAA' or 'ORD'	**No —** model determines	Neither
"	"	Quantity	Quantity	'FIX'	**Yes —** model will buy up from ITASSE quantity to NEGLVquantity	Both
"	"	Quantity	Quantity	'NOP'	**No —** Neglv is a ceiling. The model may buy up from ITASSE to NEGLV	Both

[1]Blank cells in the matrix indicate that the respective component data input field does not affect the model ou

[2]Having an impact on availability indicates that the component is included in the availability calculation.

[3]Having an impact on budget indicates that the cost of the component is included in the total cost of the sp package.

Table 3-5 also combines Tables 3-1, 3-2, and 3-3 into one table for a complete summary of starting asset and pre-specified buy combinations.

Table 3-5.
The Interaction Between Starting Assets and Pre-specified Buys — Example Impacts on Spares Targets and Buy Totals

Stock option field		Component data input fields[1]			Spares quantity	Spares quantity
Include Starting Assets?	**Use Pre-specified Buy Quantity?**	**Starting assets (ITASSE)**	**NEGLV (user specified target quantity)**	**NOP**	**Spares target**	**Buy Total**
Yes	**Yes**	2	3	'AAA'	>=3	>=1
Yes	**Yes**	2	3	'ORD'	>=5	>=3
Yes	**Yes**	2	3	'FIX'	3	1
Yes	**Yes**	2	3	'NOP'	3	1
Yes	**No**	2	3	'AAA' or 'ORD'	>=2	>=0
Yes	**No**	2	3	'FIX'	3	1
Yes	**No**	2	3	'NOP'	<=3	<=1

Note: Spares Target = Buy total - ITASSE

Force Buy Based On Pipeline % Below

You can also force the model to procure spares quantities based upon an item's pipeline, a useful procedure if you want to ensure that the item has some minimum level based upon item characteristics.

LRU % on first day — This is the percentage of the LRU pipeline (either peak or on the first analysis day, as specified by the **Purchase peak pipelines or max thru a given day** value) that the model buys sacrosanct to meet the requirement for the first analysis day. A value of 100 buys the whole pipeline, 50 buys half the pipeline, etc. An entry of 'ITEM' instructs the model to use the item-specific percentage as coded on the component data record. An entry of 'QPA' buys the pipeline for all items with a QPA greater than 2. Enter value as a percentage (e.g., 50% entered as 50), as 'ITEM,' or as 'QPA.'

SRU % on first day — This is the percentage of the SRU pipeline (either peak or on the first analysis day, as specified by the **Purchase peak pipelines or max thru a given day** value) that the model buys sacrosanct to meet the requirement of the first analysis day. A value of 100 buys the whole pipeline, 50 buys half the pipeline, etc. An entry of 'ITEM' instructs the model to use the item-specific percentage as coded on the component data record. Enter value as a percentage (e.g., 50% entered as 50) or as 'ITEM.'

LRU % on second day — This is the percentage of the LRU pipeline (either peak or on the second analysis day, as specified by the **Purchase peak pipelines or max thru a given day** value) that the model buys sacrosanct to meet the requirement for the second analysis day. As for the first day, entervalue as a percentage (e.g., 50% entered as 50), as 'ITEM,' or as 'QPA.'

SRU % on second day — This is the percentage of the SRU pipeline (either peak or on the second analysis day, as specified by the **Purchase peak pipelines or max thru a given day** value) that the model buys sacrosanct to meet the requirement of the second analysis day. As before, enter value as a percentage (e.g., 50% entered as 50) or as 'ITEM.'

Purchase peak pipelines (T, F) or max through a given day — Specifies whether the sacrosanct pipeline buys should be applied to the largest calculated pipelines over the entire analysis period, to the calculated pipelines on the actual day of analysis, or otherwise.

- A value of 'F' instructs the model to use the pipelines on the actual day of analysis.

- A value of 'T' instructs the model to use the largest (peak) pipeline up to the second analysis day. When you select 'T,' the model computes the pipeline through each day to be analyzed for each item. The model purchases the pipeline for each item sacrosanct before conducting the marginal analysis for the kit. (The day on which each item attains its peak pipeline varies according to the respective item's resupply / repair parameters.)

– Optionally, a specific day may be entered, as an integer from 0 to 99, to force the model to use the maximum pipeline values through that day.

Resupply

Resupply options (Figure 3-5) are accessible via the middle section of the Pipeline, Resupply & Options Screen (see Figure 3-1). This Resupply section enables you to adjust run time parameters that affect when base and depot repair begin in wartime for RR (remove and replace) LRUs, RRR (remove, repair, and replace) LRUs, and SRUs. You specify in the items maintenance concept (MAINTCON) field whether an LRU is RR or RRR . The standard use of those categories assumes that RRR items have repair start early in the war, while RR items have no repair until later in the war. However, you may designate LRUs as either RR or RRR on the basis of your own categorization separating them into any two groups differentiated by having their repair start on different days of the war.

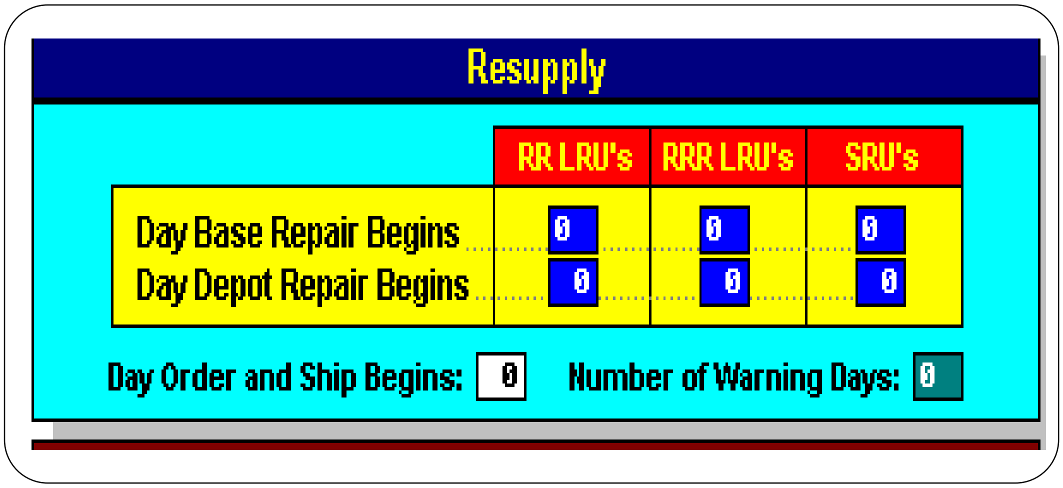


Figure 3-5
Resupply

Day Base Or Depot Repair Begins

These three values indicate the day that base repair (top row) or depot repair starts for each type of component during wartime. In each case, a value greater than the analysis day denotes no repair of that category (e.g., for a Day 30 run, '3' indicates repair will start on Day 3, while '31' indicates that no repair will be performed).

RR LRUs — First day of repair for RR LRUs.

RRR LRUs — First day of repair for RRR LRUs.

SRUs— First day of repair for SRUs.

Day Order And Ship Begins

The day when forward transportation from the depot starts. As an example, the IAF assumes that at the start of the war trucks will not be available to ship spares from the depot to the base because they will be in use for higher priority missions. If those trucks will not be available until Day 5 of the war, then this parameter is set to 5.

Number Of Warning Days

The number of days of warning before the start of the scenario (normally set to 0). In a model run that includes peacetime (Day 0), this field indicates how many days the model will use wartime resupply values before the wartime scenario actually begins.

Other Options

Other Options (Figure 3-6) are accessible via the bottom section of the Pipeline, Resupply & Options Screen(see Figure 3-1). The Other Options section enables you to adjust the remaining miscellaneous parameters that are not included in the previous two screens.

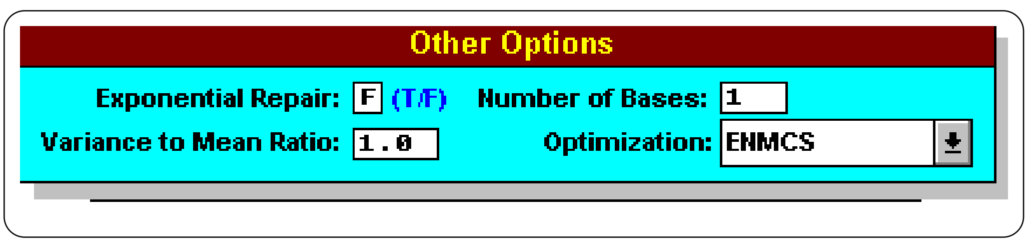


Figure 3-6
Other Options

Exponential Repair

Exponential Repair (True or False) — Specifies whether war repair times are exponentially distributed ('T') or fixed ('F'). If 'F', all repairs are deterministic and take exactly the repair time set in the component data. If 'T',the repair time for a specific item varies according to an exponential distribution with the mean equal to the item's repair time set in the component data. The main difference between deterministic and exponential repair occurs in the early part of the war, when conditions have not yet reached a steady state.

Variance-To -Mean Ratio

The variance-to-mean ratio (VMR) is a parameter used to adjust an item's demand uncertainty. The higher the value, the greater the uncertainty, and the greater the spares requirement needed to meet the target, all else being equal. From this field, the VMR is set for all component demands. The default VMR of "1.00" indicates a random or Poisson process. Sometimes actual demands can be more erratic than those generated by a Poisson process, or we may want to take into account the uncertainty about demand rates that is caused by inherent difficulty in forecasting the future. In either case, we should use a probability distribution that has more variance than Poisson; i.e., VMRs greater than 1.00. For those cases, the model uses a negative-binomial distribution with a variance larger than the mean. Currently, ISAAC will accept VMRs from 1.00 to 7.00 only (i.e., Poisson and negative-binomial probability distributions for component demand).

Number Of Bases

The number of identical (uniform) bases in the scenario must be an integer from 1 to 999. For initial provisioning, the default number of bases is 1, since we assume that all aircraft being delivered are located at one base. As the number of bases increases, so does the spares requirements needed to support the aircraft, all else being equal.

Optimization

ISAAC has three methods for producing spares mixes under cannibalization. Essentially, these correspond to three different objective functions — confidence, ENMCS, and EBO/ENMCS:

Confidence optimization maximizes the probability that the number of planes NMCS will not exceed a given target *D*. This probability is often called the confidence of meeting the target *D;* marginal analysis to maximize this confidence is called "confidence optimization."

ENMCS "optimization" considers the confidence for all possible NMCS cases (from zero on up) and optimizes a weighted sum of them. Unfortunately, ENMCS is optimized only for the point on the curve where ENMCS equals the target. Thus, if you enter an NMCS target and a budget constraint and the resulting ENMCS is significantly different from the NMCS target, you need to rerun the model with a new NMCS target equal to the ENMCS result. Nevertheless, ISAAC produces the least expensive spares mix needed to reach the ENMCS target. By experimenting with the target, ISAAC can also compute the spares mix that gives the lowest possible ENMCS for a specified cost. While maximizing confidence is not identical to minimizing ENMCS, the spares mix and performance results are not strikingly different. A spares mix built using confidence optimization normally yields a slightly better (higher) confidence level output but also a slightly worse (higher) ENMCS output and a worse (higher) EBO output.

EBO / ENMCS "optimization" uses two measures of performance: the primary measure — ENMCS — and a secondary one — EBOs. (This is a research method that has not yet been fully tested.) Though ENMCS optimization with cannibalization produces the maximum availability, it significantly increases maintenance workload, since the shop workers no longer get all spares from existing inventory (they must remove the cannibalizable parts from the failed assemblies). The spares that are cannibalized cause additional backorders and require the additional steps (and risk) involved in borrowing items from AOG aircraft and then reinstalling them later when the backorder is filled. Thus, the EBOs (included in the optimization with ENMCS) can be considered proportional to maintenance workload. The model then develops a compromise spares mix that produces slightly worse ENMCS output but significantly better EBO output (i.e., reduced maintenance workload) when compared to a pure ENMCS optimization.

Chapter 4

Developing Input Kit Data

So far we have run the model from previous “Kit” information. This chapter will describe how to develop that information. First, a kit is the input information for ISAAC — the parameters (targets, assumptions, and the scenario (total aircraft flying hours for war and non-war conditions)) and the item information (LRUs and SRUs failure rates, repair times, etc.). Currently, ISAAC has six ways to develop a kit, with the focus on developing the large amount of item information. The six ways include two distinct import file formats (worksheet and text). In addition, ISAAC gives you the capability to export kit information in each of these two file formats. If you click on **Kit** from the main menu, the following pull-down menu appears:

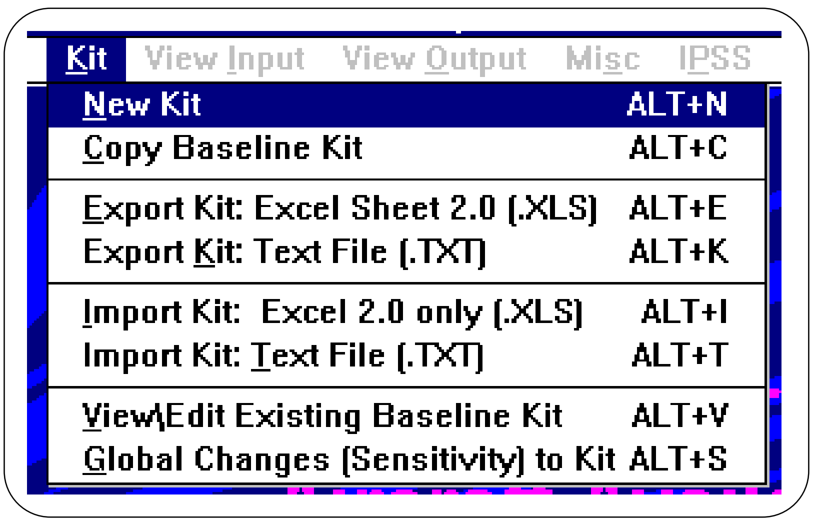


Figure 4-1.
Kit Pull Down Menu

The eight kit menu choices are described as follows. Each will be described more fully in later sections of this chapter:

– **New Kit** — This method provides a kit with default global parameters. The component data consists of one LRU record with default item level parameters. The user then enters additional item records one record at a time.

– **Copy Baseline Kit** — This method lets you select an existing baseline kit and make a copy of it. The copied kit will have the same global parameters as the kit it is copied from. The component records will be exact duplicates of the component records of the copied kit.

– **Export Kit: Excel Sheet 2.0 (.XLS) —** This selection gives you the ability to export kit information as Excel 2.0 worksheet files (*.xls) for work in Excel. ISAAC creates two separate Database Table files (*.dbf) files during the export process: one containing the kit parameter information (*_p.dbf) and one containing the kit component data (*_k.dbf).

– **Export Kit: Text File (.TXT) —** This selection gives you the ability to export kit information as System Data Format (SDF) text files (*.txt) for work in text editors. ISAAC creates two separate text files (*.txt) files during the export process: one containing the kit parameter information (*_p.txt) and one containing the kit component data (*_k.txt).

– **Import Kit: Excel Sheet 2.0 (.XLS) —** This selection enables you to import kit information in Excel 2.0 worksheet file (*.xls) format. You import the component data from a specified format .xls file (*_k.xls) where each item's data is in a single record. You are given the option of importing the corresponding kit parameter file (*_p.dbf), another kit parameter file ('*'_p.dbf) or choosing the default kit parameter file. Any item records that have invalid field values will have those values replaced with component default field values.

– **Import Kit: Text File (.TXT) —** This selection enables you to import kit information in System Data Format (SDF) text files (*.txt) format. You import the component data from a specified format .txt file (*_k.txt) where each item's data is in a single record. You are given the option of importing the corresponding kit parameter file (*_p.txt), another kit parameter file ('*'_p.txt) or choosing the default kit parameter file. Any item records that have invalid field values will have those values replaced with component default field values.

– **View\Edit Existing Baseline Kit —** This method lets you view an existing baseline kit (that is, one that was created either by you or another user.) You may also edit an existing kit with this selection. However, if you edit an existing kit after you have made model runs with the kit, you will experience inconsistencies in your model output. That is because ISAAC does not save the kit information for each run but only stores one copy of the data base for all runs. If you need to edit a kit with previous runs, you should either make a copy of the kit with a new name or edit the kit information and delete the previous runs.

– **Global Changes (Sensitivity) to Kit —** This method lets you select an existing baseline kit, make a copy of it and then make changes to multiple records at a time. The copied kit will have the same global parameters as the kit it is copied from. The component records will initially be exact duplicates of the component records of the copied kit. You can make global changes to all records (e.g., increase all base repair times by 10%) or make changes to records that meet user specified filter conditions (e.g., increase only those base repairs times for items with costs greater than $200,000).

With each of the above methods for creating a new kit, you may add, copy, modify or delete individual component records. (This will be described in detail in the Component Data section of this chapter.)

We link the parameters and item information into a kit to help ensure model output consistency. Each kit contains all the required model input necessary to run the model. That is by giving procurement team members the kit data and the same parameters, the IAF can ensure each team member develops a consistent spares mix based upon similar assumptions.

Structure of Chapter 4

Chapter 4 is organized into six sections. The first section describes each of the kit pull-down menu options that allow a user to create a new kit, modify an existing kit or export kit information. Eventually, all options (except the export options) bring you to the same place, viewing and editing input data. The remaining sections describe the common aspects of all the options. The second section describes the kit parameters on the Kit Parameters Screen that are unique to the kit (the model user is referred to Chapters 2 and 3 for specific information on the parameters that are present on both the Model Parameters Screen and the Kit Parameters Screen). The third section describes the Kit Component Data Screen, along with the component input parameters, that is used to enter or modify existing item level information. The fourth section discusses the warnings generated by invalid field values. Included in this section are the valid range of field values for each of the kit component level fields and the difference between the warning and error windows. The fifth section describes how the model develops the indenture structure of the weapon system under consideration. The last section describes how to make global modifications to kits using the sensitivity change kit option.

Kit Pull-Down Menu Options

Each of the eight kit pull-down menu options will be described in this section. The focus is on creating a new kit or modifying an existing kit. Information on creating new component records or modifying existing component records will be presented in the next section "Component Data."

New Kit

When you select **New Kit** from the **Kit** pull down menu, you will see the Define Kit ID dialog box (Figure 4-2).

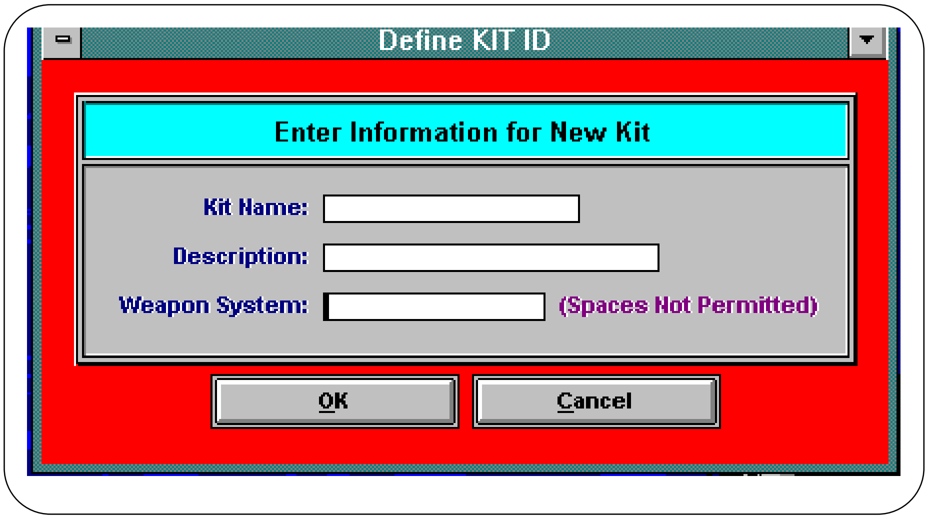


Figure 4-2.
Define Kit ID Dialog Box

You must enter a **Kit Name, Description** and **Weapon System** for the new kit. Your (kit) **Description** must be unique and should be meaningful to you so as to aid in future identification of this kit. The **Weapon System** is used in the model to identify the kit information. ISAAC will automatically insert the weapon system name in the Next Higher Assembly field of the LRU component record. Do not leave the **Weapon System** blank!

Once you have entered the information select **OK** and the Kit Parameters Screen will appear (Figure 4-3). The information displayed on the Kit Parameters Screen as well as the information that may be accessed through the **Scenario Baseline** or **Stock / Resupply / Other Options** buttons are the default parameter values. Each of the global parameter fields has a valid field value that the model will accept.

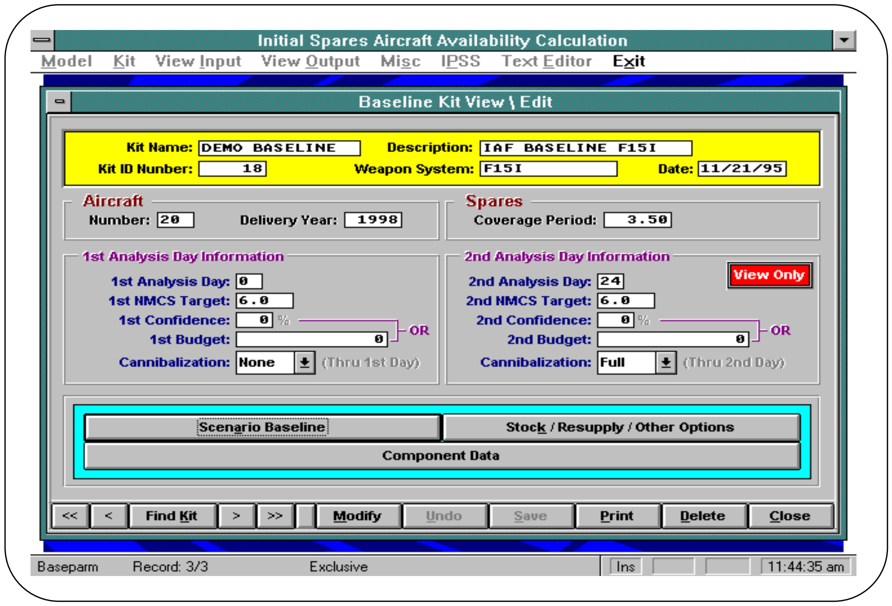


Figure 4-3.
Kit Parameters Screen

– If you want to change information on the Kit Parameters Screen, you must click the **Modify** button to be able to edit the screen.

– You can then edit any of the kit parameter data. Once you are done with the top part of the screen, Scenario Screen, and Stock / Resupply / Other Options Screen, click on **Save**.

► See Chapter 2 for information on the Scenario Screen.

► See Chapter 3 for information on the Stock/ Resupply/ Other Options Screen.

You only need to change those global parameters that you would like to change on the Kit Parameters Screen, Scenario Screen and Stock, Resupply & Other Options Screen.

You must enter item level information via the Kit Component Data Screen. To access this window click the **Component Data** button. You can only edit the component data information when the **Component Data** button is highlighted. This button is only highlighted when the Kit Parameters Screen is in the 'View' mode (not the 'Modify' mode). The screen in Figure 4-4 should appear.

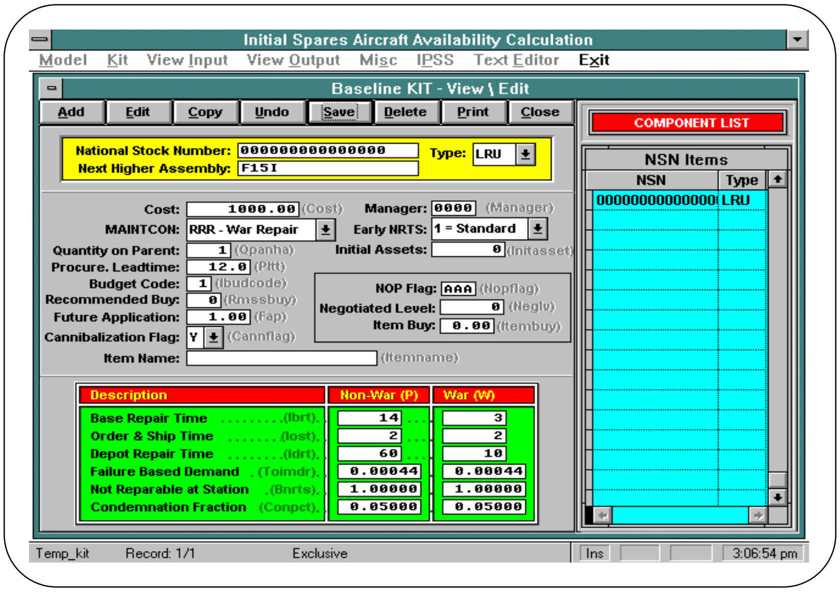


Figure 44.
New Kit Component Data Screen

The only record in the Kit Component Data Screen, for a new kit, is the default record for an LRU with National Stock Number 00000000000000. The item level parameters displayed for this record are the ISAAC item level default values. These default values were designed to speed up your ability to populate the item level database from scratch.

See the Component Data section, later in this chapter, for details on adding, copying or editing specific component data. Once you have entered all item level information, click the **Close** button to return to the Kit Parameters Screen.

– You click on the **Close** button of the Kit Parameters Screen when you are done editing.

Copy Baseline Kit

If you select **Copy Baseline Kit** from the **Kit** pull-down menu, ISAAC copies the parameters of an existing kit and the item data. Your first step is to select the kit you want to copy (see Figure 4-5).

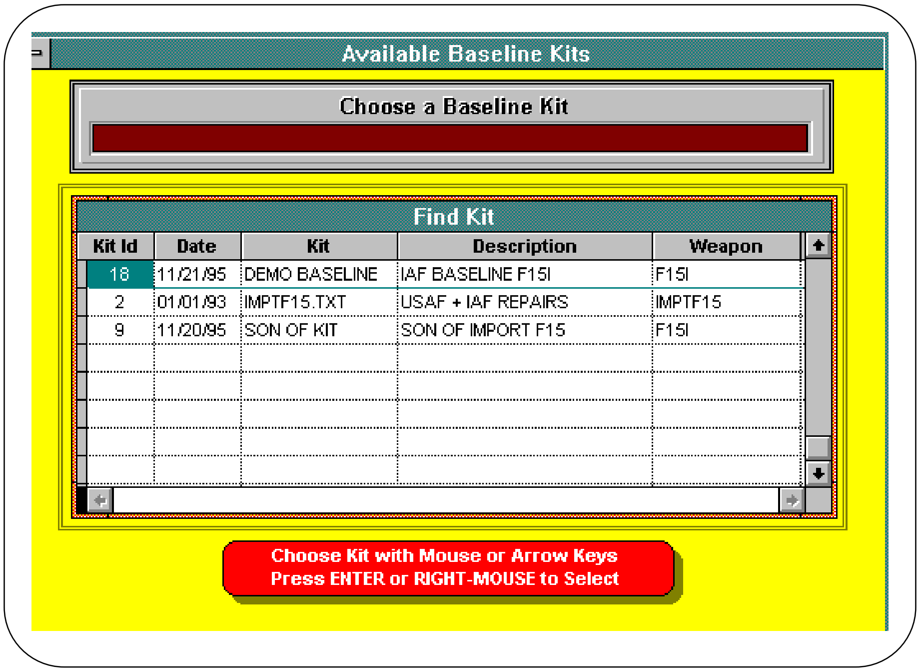


Figure 4-5.
Choose a Baseline Kit

Move the cursor to the kit you would like to copy and then select it by pressing **'Enter'** or by clicking the right mouse button. The Define Kit ID dialog box (Figure 4-2) should then appear. You must enter a **Kit Name, Description** and **Weapon System** for the new kit. Your (kit) **Description** must be unique and should be meaningful to you so as to aid in future identification of this kit. The **Weapon System** is used in the model to identify the kit information. ISAAC will automatically insert the weapon system name in the Next Higher Assembly field of the LRU component record. Do not leave the **Weapon System** blank!

Once you have entered the information select **OK** and the Kit Parameters Screen will appear (Figure 4-3). The information displayed on the Kit Parameters Screen as well as all other windows that can be accessed from this screen (through the **Scenario Baseline**, **Stock / Resupply / Other Options**, and **Component Data** buttons) is an exact duplicate of the baseline kit that this was copied from.

– If you want to change information on the Kit Parameters Screen, you must click the **Modify** button to be able to edit the screen.

– You can then edit any of the kit parameter data. Once you are done with the top part of the screen, Scenario Screen, and Stock / Resupply / Other Options Screen, click on **Save**.

- If you want to change item level information (add, delete or modify) you must click on the **Component Data** button. You can only view or edit the component data information when the **Component Data** button is highlighted. This button is only highlighted when the Kit Parameters Screen is in the 'View' mode (not the 'Modify' mode).

- You are then able to edit any of the *item* information in the kit. See the Component Data section, later in this chapter, for details on editing specific component data. At this stage be careful that the data you enter is correct since it is being saved as it is entered. To close the Kit Component Data Screen you must click on the **Close** button near the top of the window.

- You click on the **Close** button of the Kit Parameters Screen when you are done editing.

Export Kit

ISAAC gives you the ability to export kit information in either of two different file formats. Excel worksheet files (*.XLS) for work in Excel worksheets. System Data Format (SDF) text files (*.TXT) for work in text editors. ISAAC also enables you to import kit information from the same two file formats.

ISAAC creates two separate files during the export process: one containing the kit parameter information and one containing the kit component data. The export process consists of the following three distinct steps:

- Select either **Export Kit: Excel Sheet 2.0 (.XLS)** or **Export Kit: Text File (.TXT)** from the **Kit** pull down menu.

- Choose a Baseline Kit from the Available Baseline Kits window that is displayed (see Figure 4-5). Once you select a kit Figure 4-6 will be displayed.

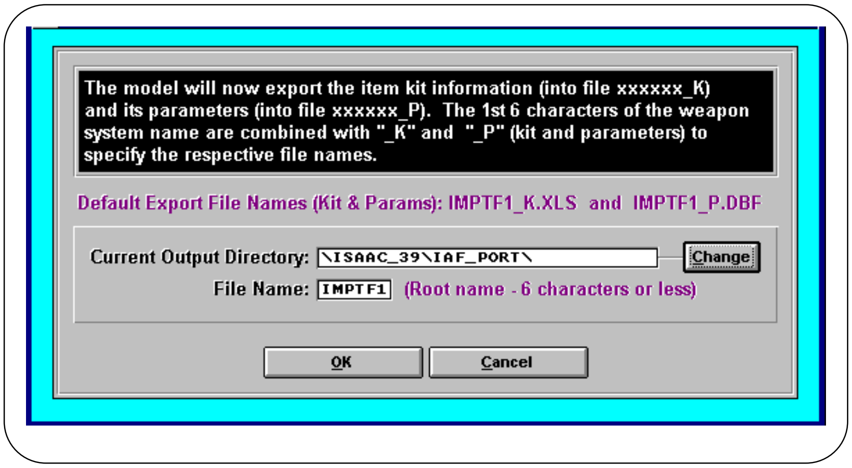


Figure 4-6.
Export Kit and Parameters Dialog Box

– Select the output directory and file name. The default output directory is the IAF_PORT subdirectory of your ISAAC root directory. (If you want to export to a different directory or subdirectory select the **Change** button. Choose the directory / subdirectory that you want to export the kit information to and **Select** it.) The default file name is the first 6 characters of the **Weapon System** field from the kit that you are exporting. The default export file names are displayed in purple above the **Current Output Directory** text box. (If you want to create your own simply type in your filename into the **File Name** text box.) When you are finished select the **OK** button:

► The model will export the item kit information into a file using the filename you selected (up to six characters) followed with a _k and the appropriate file extension (e.g., xxxxxx_k.xls or xxxxxx_k.txt).

► The model will export the kit parameter information into a file using the filename you selected (up to six characters) followed with a _p and the appropriate file extension (e.g., xxxxxx_p.dbf or xxxxxx_p.txt).

You are now back at the initial ISAAC screen.

[Working with Kit Information Exported in Excel Format]

When you export a kit in Excel format, the kit information is exported in an Excel 2.0 worksheet format. This option enables you to modify the kit information in the familiar Excel worksheet. The kit that you exported can be opened in any version of Microsoft Excel. If you are using a more recent version of Microsoft Excel than the 2.0 version, you will see the following message when you **Save** the file: "'XXXX_K.XLS' was created in a previous version of Microsoft Excel. Do you want to update it to Microsoft Excel [current version] format?" Choose the **No** button. ISAAC can only import Excel files that are in the Excel worksheet format. If the file is in a more recent version of Excel or in a workbook format and you attempt to import it into ISAAC, you will get an "Invalid Excel file format" program error.

Import Kit

The import kit information process is essentially the reverse of the export kit information process. ISAAC gives you the ability to import kit information in either of two different file formats. Excel worksheet files (*.XLS) for work in Excel worksheets and System Data Format (SDF) text files (*.TXT) for work in text editors.

ISAAC creates two separate files during the export process: one containing the kit parameter information and one containing the kit component data. When you import a kit ISAAC lets you import these same two files. You first select which kit component data you want to import. At that point you are given a choice, you can import a parameters information file or use the ISAAC default parameters. We will describe importing the parameters file in more detail, later in this chapter.

ISAAC will use the imported kit component data file to create the component records for the new kit. If there are any invalid field entries, these values will be replaced by the field default values. Any replacement of invalid field data with default values is considered to be a correction of an error and will be printed to an error file that will be displayed during the kit creation process.

ISAAC is structured to import information directly from the IPSS (Initial Provisioning Support System) using the .TXT option. We assume the IPSS model generated this information so it is in the proper format. However, if the IAF develops an item database in another software package (e.g., Lotus 1-2-3, Microsoft Excel, or Microsoft FoxPro), it must be in the field format given in table 4-1. To get the proper structure, export a Excel kit with ISAAC, open the file with the other software package, deleted the data, and use the remaining structure. Whatever software package is used to create the kit database, the file must be saved as a Microsoft Excel Spreadsheet (XLS). In Excel you can save it as a version 2.0 through 4.0 spreadsheet file. In the other software packages, save it as an **Excel Worksheet (.xls)** in Lotus 1-2-3 or as an **Excel Worksheet (.XLS)** in Microsoft FoxPro. It is important that you save it as an Excel Worksheet NOT as an Excel Workbook.

Table 4-1.
Import Text file Structure

Field No.	Model name abbreviated	Full kit field name	ISAAC field type	Field width	Decimal	Start col	End col
1	RTYPE	Type	CHAR.	4	0	1	4
2	NSN	National Stock Number	CHAR.	18	0	5	22
3	COST	Cost	NUM.	11	2	23	33
4	IQPA	Quantity Per Aircraft	NUM.	4	0	34	37
5	QPANHA	Quantity on Parent	NUM.	4	0	38	41
6	FAP	Future Application	NUM.	6	2	42	47
7	PLTT	PLTT	NUM.	6	1	48	53
8	INITASSET	Init Asset	NUM.	7	0	54	60
9	FIL1		CHAR.	1	0	61	61
10	NHANSN	Next Higher Assembly	CHAR.	18	0	62	79
11	IBUDCODE	Budcode	NUM.	2	0	80	81
12	NEGLV	Negotiated Level	NUM.	6	0	82	87
13	FIL2		CHAR.	1	0	88	88
14	MAINTCON	MAINTCON	CHAR.	3	0	89	91
15	ITEMBUY	Item Buy	NUM.	5	2	92	96
16	FIL3		CHAR.	1	0	97	97
17	CANNFLAG	Cannibalization Flag	CHAR.	1	0	98	98
18	FIL4		CHAR.	1	0	99	99
19	NOPFLAG	NOPFLAG	CHAR.	3	0	100	102
20	EARLYPCT	Early NRTS	NUM.	5	2	103	107
21	IBRTP	Base Repair Time (Peace)	NUM.	6	0	108	113
22	IBRTW	Base Repair Time (War)	NUM.	6	0	114	119
23	IOSTP	Order and Ship Time (Peace)	NUM.	6	0	120	125
24	IOSTW	Order and Ship Time (War)	NUM.	6	0	126	131
25	IDRTP	Depot Repair Time (Peace)	NUM.	6	0	132	137
26	IDRTW	Depot Repair Time (War)	NUM.	6	0	138	143
27	TOIMDRP	Failure Based Demand (Peace)	NUM.	8	5	144	151
28	TOIMDTW	Failure Based Demand (War)	NUM.	8	5	152	159
29	BNRTSP	Not Reparable at Station (Peace)	NUM.	8	5	160	167
30	BNRTSW	Not Reparable at Station (War)	NUM.	8	5	168	175
31	CONPCTP	Condemnation Fraction (Peace)	NUM.	8	5	176	183
32	CONPCTW	Condemnation Fraction (War)	NUM.	8	5	184	191
33	CLASS	Classification of the Component	CHAR.	1	0	192	192
34	ITEMNAME	Item Name	CHAR.	19	0	193	211
35	IAFID	IAF Part Number	CHAR.	9	0	212	220
36	USNSN	National Stock Number	CHAR.	15	0	221	235
37	MANAGER	Manager	CHAR.	4	0	236	239
38	MRU	Minimum Replacement Quantity	NUM.	3	0	240	242
39	RMSSL	Manufacturer’s Rec Min Stk Level	NUM.	3	0	243	245
40	RMSSBUY	Manufacturer’s Rec Min Buy Qnt	NUM.	3	0	246	248
41	WPNSYS	Weapon System	CHAR.	15	0	249	263
42	SCHMAINT	Scheduled Maintenance	NUM.	8	5	264	271

Common Components

ISAAC also considers the impact of "common components" and that requires additional imported information. Common components are components common to both the initial provisioning aircraft and other IAF aircraft. An example would be a component on the initial provisioning aircraft series (e.g., F-15I) and also on the F-15C and the F-16D. Treatment of common components must apply any surplus stock already in the IAF inventory toward the requirement and must consider economies of scale. Since stock is already available for other aircraft, the new aircraft needs less inventory than if the part were not common. ISAAC uses a simple approximation to incorporate those benefits that are expressed in terms of "free assets": assets available to the item free of charge because of its commonality characteristics (See the *Optimizing Spares Support: The Aircraft Sustainability Model* for more information).

To calculate free assets, the IAF supplies information relating to the common components characteristics from the IAF replenishment data base (DB) in Israel called MALHA. The IPSS also develops this common component replenishment DB information. The replenishment information is imported in the same file and in the same format as the initial provisioning aircraft data (Table 4-1) with three differences:

– Replenishment component data has the word "OTHER" in the weapon system field ("Wpnsys"),

– Replenishment component data assumes demand (Toimdrp) equals the total demands from all aircraft and all items per day (as opposed to the model's standard definition which is the demand per flying hour for a single item),

– Replenishment component data assumes assets (InitAsset) equal the current IAF inventory position.

Though all fields in Table 4-1 can have information, for the common component calculation the following fields are required: NSN, PLTT, INITASSET, IBRTP, IOSTP, IDRTP, TOIMDRP, BNRTSP, CONPCTP or field numbers 2, 7, 8, 21,23,25,27,29,31 in Table 4-1.

Steps for Importing a Kit

Once you select either **Import Kit Excel Sheet 2.0 (.XLS)** or **Import Kit: Text File (.TXT)** from the **Kit** pull down menu, you should see an Open dialog box similar to Figure 4-7.

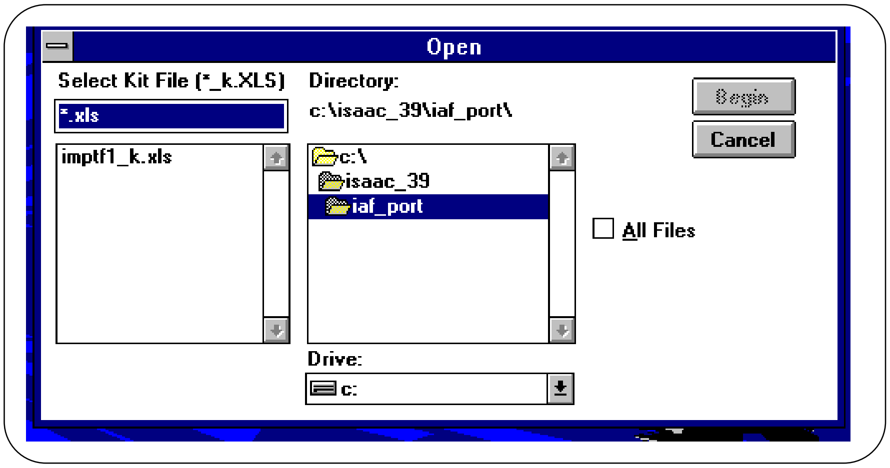


Figure 4-7.
Select Kit File Dialog Box

The default directory for the import routine is the primary ISAAC directory. Text files to be imported should generally be stored in the iaf_port subdirectory. To select the file to import, perform the following steps (as applicable):

– If the file you want to open is on a different drive, select the drive you want from the Drive box.

– If the file you want to open is on the same drive but in a different directory or subdirectory, select the directory and or subdirectory you want from the Directories box. Double-click the directory, or press the up or down arrow key to select the directory, and then press **'Enter'**.

– ISAAC displays the names of all files in that directory that are of the type selected in the Open file text box. The default file extension for the file to be imported is ".txt " although ISAAC accepts any extension.

– From the list of files, select the file you want to open.

– Double-click the filename or choose the **Begin** button.

It is important to remember that although ISAAC has some filter checks that it employs when importing data files these checks are not foolproof. It is important to import kit information and kit parameter files that are in the proper format (all files that ISAAC exports are in the proper format for subsequent import). It is important to only import parameters files that have been developed in and exported from ISAAC. If in doubt about the parameters file, it is best to use the default parameters file and edit it after you have imported the item data.

Once you select a file from the Select Kit File dialog box (Figure 4-7). Figure 4-8 will be displayed.

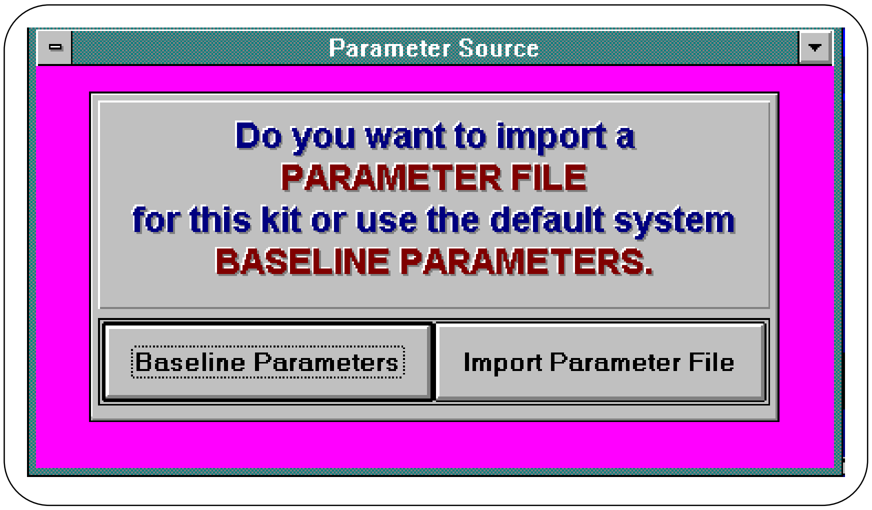


Figure 4-8.
Parameter Source Window

Choose the parameters file that you want to use. Select **Baseline Parameters** if you want to use the ISAAC default parameters. If you want a specific parameters file select **Import Parameter File** and the Select Parameter File dialog box will be displayed. Choose the appropriate parameters file (either the default or a specific parameters file).

Once you have chosen which parameters file to import, you need to perform the following steps to convert the files to an ISAAC baseline file:

- A Define Kit ID window (Figure 4-2) will be displayed. You must enter a **Kit Name, Description** and **Weapon System** for the new kit. Your (kit) **Description** must be unique and should be meaningful to you so as to aid in future identification of this kit.

- The model then looks for matches between the initial provisioning items and the replenishment DB items in order to calculate Free Assets from common components (see earlier description). A window similar to Figure 4-9 will appear. The window displays the number of replenishment items that are in the imported database and the number of item record matches found between the replenishment information and the initial provisioning information. (In our example the number of matches [4] exceeds the number of replenishment records because one match is to a common SRU in the initial provisioning data that has a separate record for each different next higher assembly). To proceed, you must choose the **Continue** button at the bottom of the window.

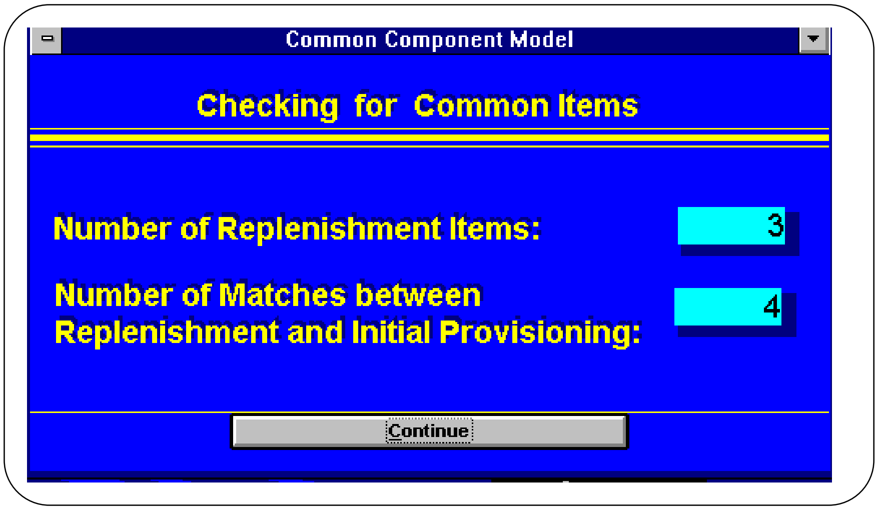


Figure 4-9.
Checking for Common Items

– At this point, ISAAC allows you to change the item level default values. A screen similar to figure 4-10 will appear. Any changes you make to these field default values will update the database (i.e., the default nsn component data values will be updated to reflect your changes).

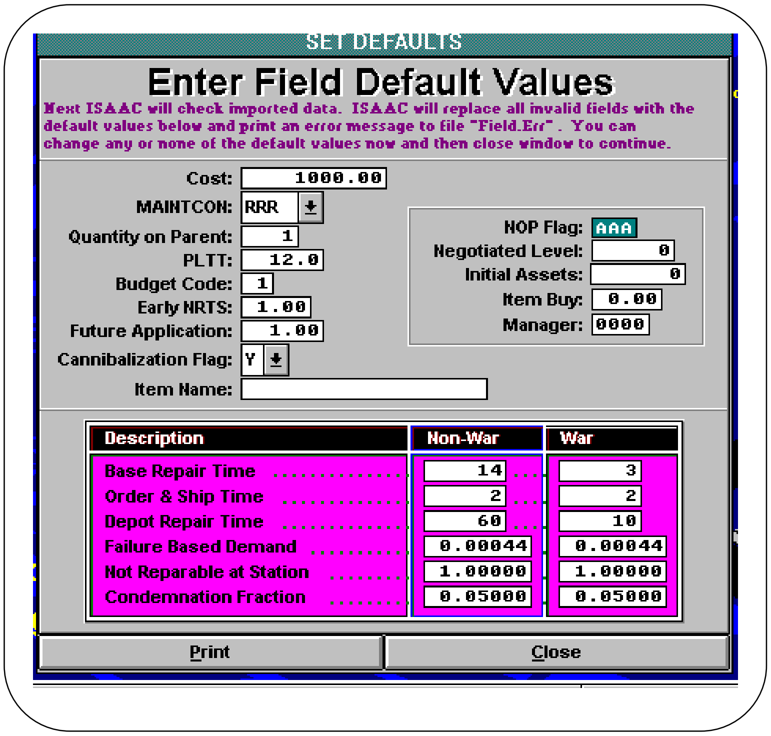


Figure 4-10.
Enter Field Default Values Screen

– Next ISAAC will check the validity of the imported data.

– ISAAC will replace all invalid field values (not within the specified valid range defined in table 4-2) with the default values that are entered on the Field Default Values Screen (Figure 4-10) and print an error message to file "Field.err." For instance, if the imported data has an item's condemnation fraction set to 2.0, which is outside the valid range from 0 to 1, then ISAAC will replace it with 0.05000 (default value from Figure 4-10).

► The field definitions are presented later in this chapter under the "Basic Fields" and the "Conditional (Wartime / Non-wartime) Fields" sections. Valid entries for each of the fields are specified in Tables 4-2 "Valid Range of Values for Component Data Fields."

– You can change any or none of the default values on this screen. Once you are through making changes (if any) to the default values, close the screen to continue model processing.

– If there are any field errors discovered during the model component field validation, a "Warning: Found Field Errors" window (Figure 4-11) will appear.

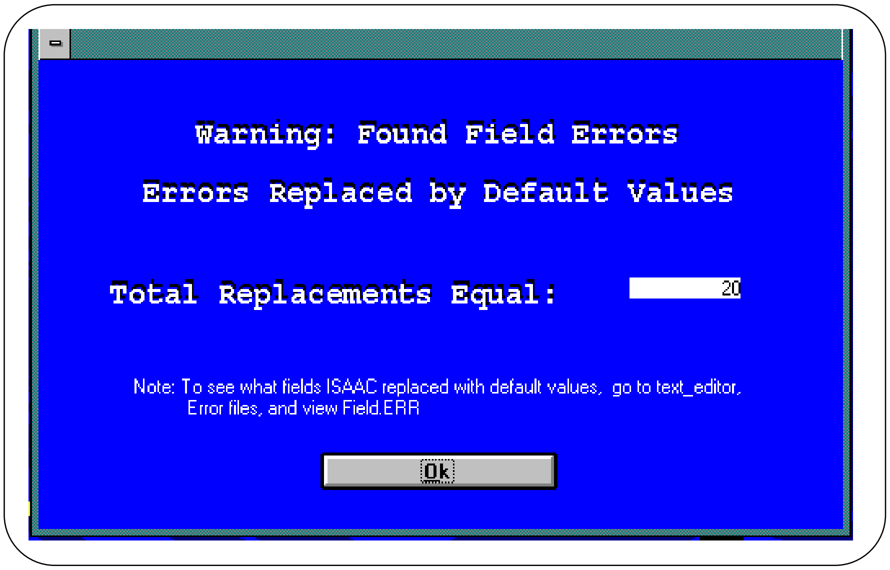


Figure 4-11.
Warning: Found Field Errors Window

– This window will indicate how many errors have been discovered and replaced with the field default values. Press **Ok** to continue.

– The actual errors and the resultant corrections can viewed by the next screen (Figure 4-12) within our text editor (for specifics on the ISAAC Text Editor see Chapter 6.)

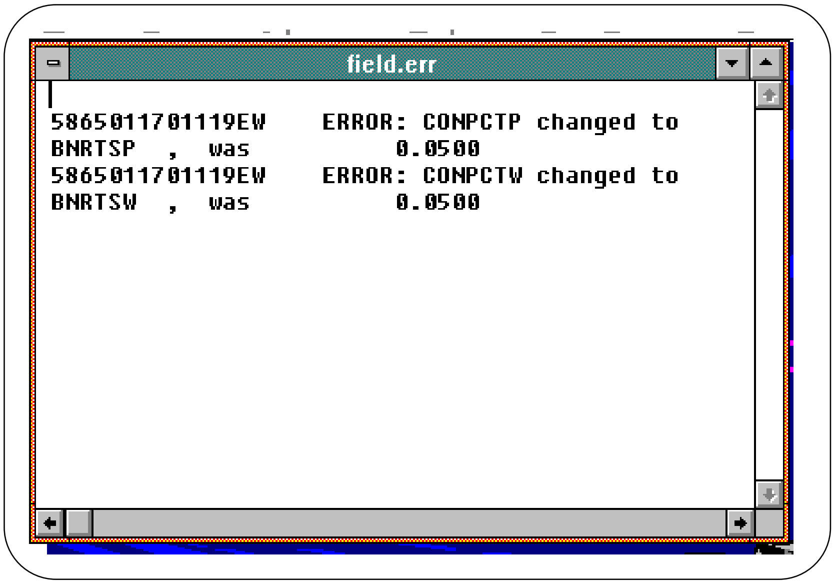


Figure 4-12.
Field Error Window

– Press the **'Esc'** key or click on the control menu box (upper left of the window) to close this window. You must close this window to continue model processing. This will complete the model indenture checking and field validation model processes.

Next you are at the Kit Parameters Screen. If you want to change information on this screen, you must click the **Modify** button to be able to edit the screen.

– You can then edit any of the kit parameter data. Once you are done with the top part of the screen, Scenario Baseline, and Stock / Resupply / Other Options click on **Save**. All the parameter, scenario, and item data fields must have valid entries before the model will allow you to save the kit parameters.

– You are then able to edit any of the component information in the kit.

– You click on **Close** when you are done editing.

View \ Edit Existing Baseline Kit

This method lets you view an existing baseline kit (i.e., one that was created either by you or another user.) You may also edit an existing kit with this selection. However, if you edit an existing kit after you have made model runs with the kit, you will create inconsistencies in your model output. That is because ISAAC does not save the kit information for each run but only stores one copy of the data base for all runs. If you need to edit a kit with previous runs, you should either make a copy of the kit (giving it a new name) or edit the kit information and delete the previous runs. To remind you of those possible problems, after you select **View\Edit Existing Baseline Kit** from the **Kit** pull-down menu Figure 4-13 will be displayed.

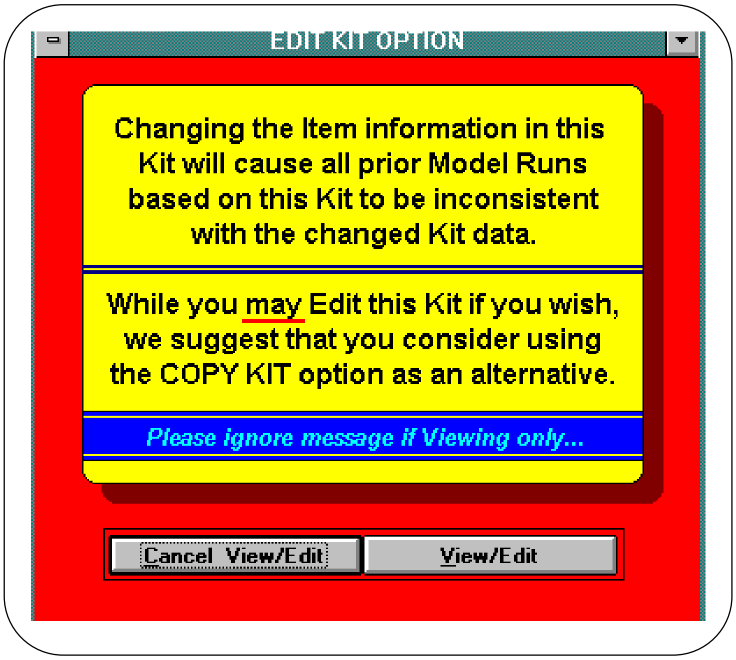


Figure 4-13.
Edit Kit Option Window

If you click on the **View/Edit** button, the Kit Parameters Screen (Figure 4-3) will be displayed for the kit at the top of your kit database. If you would like to view or edit a different kit click on the **Find Kit** button and the Available Baseline Kits window (Figure 4-5) will be presented.

Sensitivity Changes

Global Changes (Sensitivity) to Kit is a way of globally editing all item records with a single command and will be discussed at the end of this chapter after description of some of the related topics.

KIT PARAMETERS

The kit parameters are essentially the same as the model parameters described in Chapters 2 and 3. All of the global kit parameters are displayed on the Kit Parameters Screen (Figure 4- 14) or are accessed through that screen.

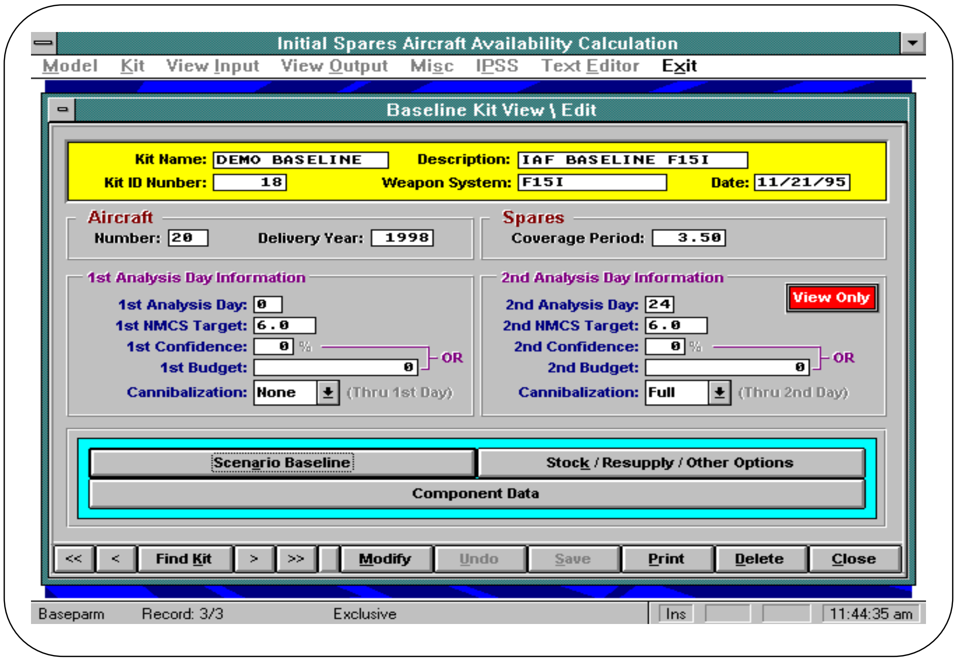


Figure 4-14.
The Kit Parameters Screen

For descriptive purposes we have divided the Kit Parameters Screen into four sections: that we will list now and describe later in this section:

1. Bottom row of buttons

2. Top section

3. Middle section

4. Lower section of buttons

In general, we will limit our description of this screen and its functions to those portions that are different or that function differently than in the Model Parameters Screen. We refer the user to Chapters 2 and 3 for detailed descriptions of those portions and functions that are the same as in the Model Parameters Screen.

Bottom Row of Buttons on the Kit Parameters Screen

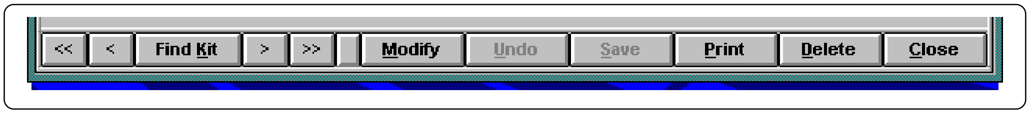


Figure 4-15.
Bottom Row of Buttons on the Kit Parameters Screen

Find Kit — The two buttons on either side of the **Find Kit** button can be used to move around the library of previous kits without viewing them through the Available Baseline Kits Window. The advantage to using these buttons is that it allows you to view the run time parameters of the previous kits in the familiar Parameters Screen format. The buttons are defined as follows:

- Selecting the **>** button will fill the Kit Parameters Screen with the global parameters from the previous kit that was immediately above the currently displayed kit in the Kit Library.

- Selecting the **>>** button will fill the Kit Parameters Screen with the global parameters from the kit at the top of the Kit Library.

- Selecting the **>** button will fill the Kit Parameters Screen with the global parameters from the previous kit that was immediately above the currently displayed kit in the Kit Library.

- Selecting the **<<** button will fill the Kit Parameters Screen with the global parameters from the kit at the bottom of the Kit Library.

Modify — Selecting this button will enable you to modify the basic and advanced global parameters.

Undo — Undoes or reverts all changed parameter values to their value before the **Modify** button was pressed.

Save — Saves any changes to the kit parameter information that have been made since the **Modify** button was pressed.

Print — This will provide access to a print menu (Not currently working). See Chapter 5 for instructions on how to print model run input and output data.

Delete — This will delete all records associated with this particular **Kit Description**.

Close — Closes the Kit Parameters Screen and returns you to the initial screen.

Top Section of Kit Parameters Screen

The Top section of the Kit Parameters Screen (Figure 4-16) consists of the kit identification fields. These fields are unique to the kit, have been either entered by the user on a previous window (see Figure 4-2) or by the model, and can not be edited.

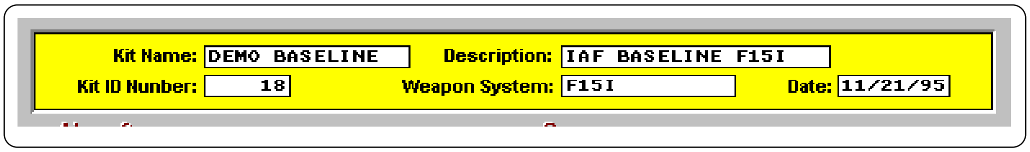


Figure 4-16.
Top Section of Kit Parameters Screen

Kit Name — The kit name is a name associated with a particular kit. This name is entered by the user during the kit creating process.

Kit ID Number — The kit ID number is a unique number assigned by ISAAC to each baseline kit for tracking and identification purposes in the library of available kits.

Description — The (kit) **Description** should be meaningful to you so as to aid in future identification of this kit. This description is entered by the user during the kit creating process.

Weapon System — The weapon system name also helps the user identify the Kit. ISAAC also uses this field during the indenture check process and inserts it as the Next Higher Assembly field in each of the LRU component records.

Date — The date that the kit was created. This date is entered by ISAAC.

Middle Section of Kit Parameters Screen

The middle section of the Kit Parameters Screen (Figure 4-17) consists of the primary global parameter fields. See Chapter 2 for information on setting the basic global parameters.

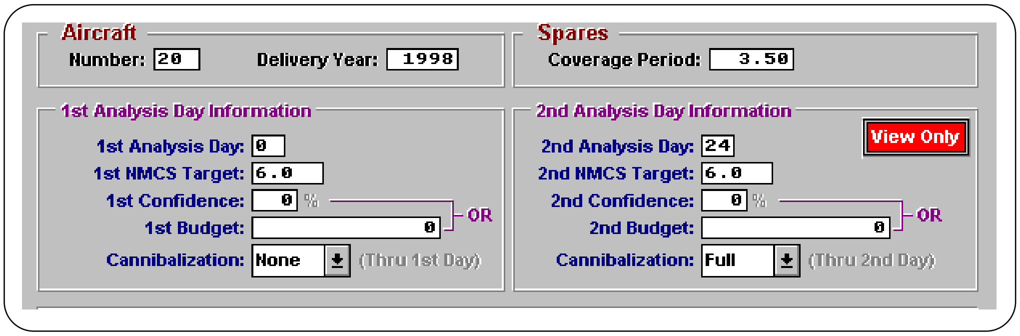


Figure 4-17.
Middle Section of the Kit Parameters Screen

Lower Section of Kit Parameters Screen

The lower section of the Kit Parameters Screen (Figure 4-18) consists of the following three buttons that will each open up another screen:

– **Scenario Baseline** — See scenario section from Chapter 2 for information on the Flying Hour Scenario screen.

– **Stock / Resupply / Other Options** — See Chapter 3 for information on setting the advanced global parameters on the Stock, Resupply & Other Options Screen.

– **Component Data** — See the next section.

The buttons are displayed below.

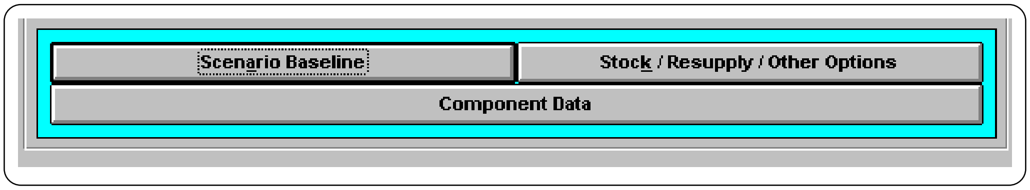


Figure 4-18.
Lower Section of Kit Parameters Screen

Component Data

The component data consists of the item information (LRU and SRU failure rates, repair times, etc.) that ISAAC processes using the **Model** option to determine which spares are required to meet the availability target(s).

For organizational purposes, we have divided the Component Data section into the following five subsections:

1. Identification fields

2. Basic fields

3. Conditional (wartime / non-wartime) fields

4. Viewing different components via the component list

5. How to modify existing or create new component information

The Kit Component Data Screen (Figure 4-19) enables you to view or edit individual component records. This Baseline Kit Component Data Screen is not intuitive. There are specific procedures for viewing, creating, modify and deleting component field information (described below).

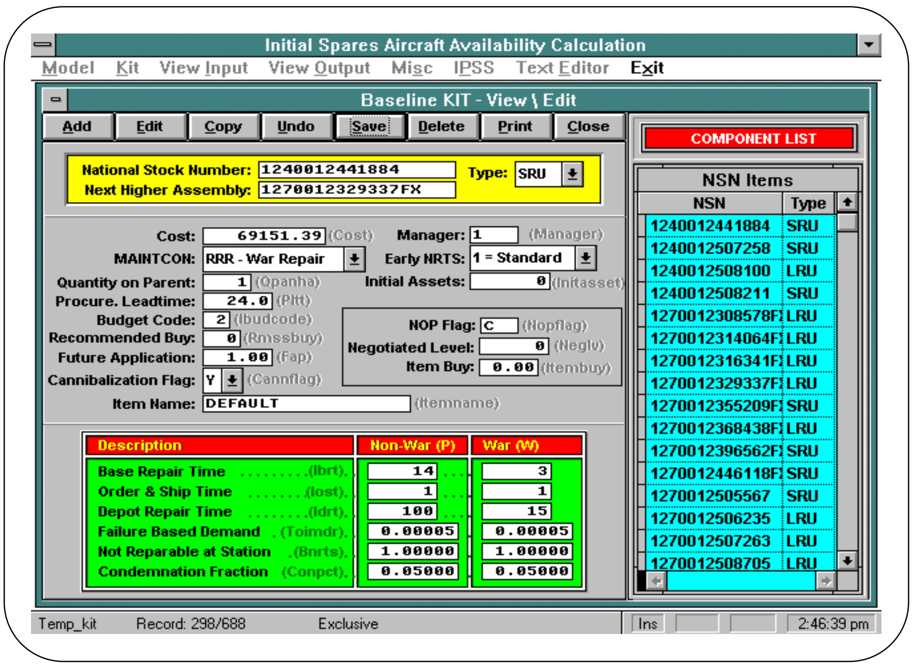


Figure 4-19.
Kit Component Data Screen

Identification Fields

Item identification information consists of three data fields that uniquely define each component. These fields are shown in Figure 4-20 and defined below.

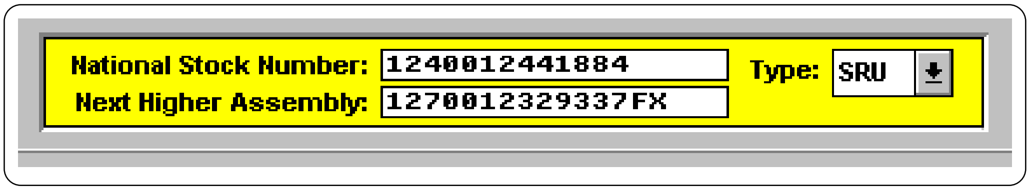


Figure 4-20.
The Item Identification Section of the Kit Component Data Screen

National Stock Number — National stock number of the component Field type: Text box; Field width: 18 character spaces.

Next Higher Assembly — Next higher assembly national stock number; the next higher assembly forLRUs will be the weapon system.Field type: Text box; Field width: 18 character spaces.

Type — The type of reparable this spare is classified as. Field type: Drop-down list box consisting of **LRU** or **SRU**.

Basic Fields

Basic field information consists of fourteen data fields that do not vary on the basis of war or nonwar conditions. These field values are also unaffected by operational parameters such as flying hours. These fields are shown in Figure 4-21 and defined below.

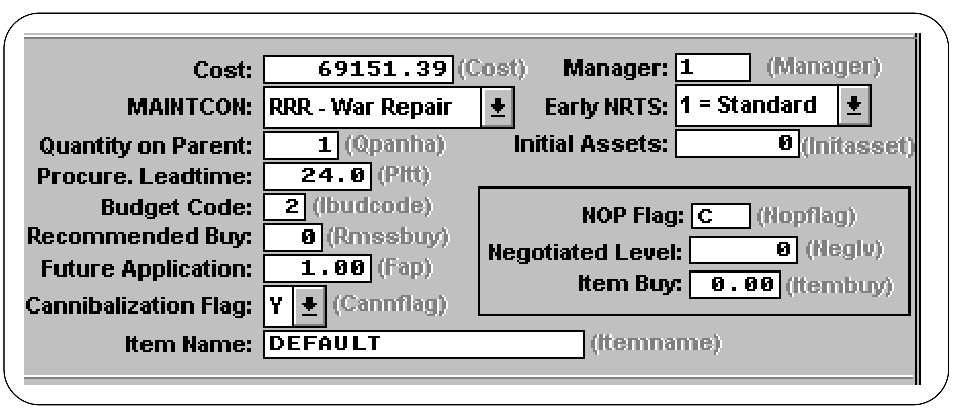


Figure 4-21.
The Fixed Component Fields Section of the Kit Component Data Screen

Valid entries for each of the following fields are depicted in Table 4-2.

Cost — Unit cost of the component in US Dollars. Field type: Text box; Field width: 11 numeric spaces (2 decimal places).

MAINTCON — This affects when (if ever) wartime base and depot repair of failed LRUs and SRUs begins. The maintenance concept is used to group spares for the purpose of establishing when repair begins for each group in wartime. Specifically the MAINTCON can be used to determine when base and depot repair begin during war for RR (remove and replace) LRUs, RRR (remove, repair, and replace) LRUs, and SRUs. The standard use of those categories assumes the RRR items have repair start early in the war and RR have no repair until later in the war. However, the user may specify an LRU as RR or RRR based upon their own definition that separates LRUs into any 2 groups of items that each have their repair start on different days of the war. The Resupply section of the Stock, Resupply and Other Options Screen enables the user to enter the day repair starts (using the starting day of the war as the reference point) at the base and at the depot. A separate repair start date can be specified for RR or RRR LRUs and SRUs at the depot and at the base. Field type: Drop-down list box consisting of **RR No War Repair** or **RRR War Repair**.

Quantity on Parent — Quantity of the component installed on each unit of the respective next higher assembly. Field type: Text box; Field width: 4 numeric spaces.

Procure. Leadtime — Procurement leadtime total for the item in months. The PLTT is the time from when an item is condemned to when a serviceable replacement for the item is procured and available at the base. The PLTT can be thought of as the sum of the administrative leadtime required to order the item once the failure is discovered, the production leadtime and the time required to process and ship the item. Field type: Text box; Field width: 6 numeric spaces (1 decimal place).

Budget Code — A user defined budget code from 1 to 99 which is used to record composite output fields for a group of items (such as different aircraft subsystems) Field type: Text box; Field width: 2 numeric spaces (index).

Recommended Buy -- The recommended manufacturer system stock buy quantity (RMSSBUY). With a standard initial procurement, this field value is what the manufacturer has recommended as a spares buy quantity. It is assumed that this field value is imported with the rest of the component level information. However, the user can use this field to evaluate any spares mix (see Chapter 8). Field type: Text box; Field width: 3 numeric spaces.

Future Application — Future application percentage (FAP): the percentage of aircraft that will be configured with this NSN. For common SRUs, this is the percentage of weapon systems that have this particular SRU on its respective LRU (Next Higher Assembly). For example, if a common SRU is on 50% of LRU A and 50% of LRU B and LRU A is on 40 percent of the weapon systems and LRU B is on 80% of the weapon systems, then the user should enter the following FAP values: 1) for common SRU record 1 (LRU A is the next higher assembly) the FAP is 0.2 (0.5x0.4); 2) for common SRU record 2 (LRU B is the next higher assembly) the FAP is 0.4 (0.5x0.8). Field type: Text box; Field width: 6 numeric spaces (2 decimal places).

Cannibalization Flag — Specifies whether an LRU can be cannibalized or not. '**N**' indicates that the item can not be easily cannibalized, while '**Y**' indicates that the item can be easily cannibalized. This field is only used when "Partial" is selected for the 'Cannibalization' field of the Parameters screen. Field type: Drop-down list box consisting of **Y** (Yes) or **N** (No); Field width: 1 character space.

Item Name — The noun nomenclature of the component part. Field type: Text box; Field width: 19 character spaces.

Manager — Item manager code. Field type: Text box; Field width: 4 character spaces.

Early NRTS — The percentage of Not Reparable at This Station which is done early (i.e., the percentage of failed items that are NRTSed directly to depot without going through the base repair cycle). A percentage of 100% (entered as **1**) means the item is sent directly to depot with no delay at base (the standard assumption used for most items). A percentage of 0% (entered as **0**) means an item waits a base repair time (isolating the problem) before being sent to the depot for repair (an item where this happens is an exception). Field type: Drop-down list box consisting of 1 = Standard or 0 = Exception.

Initial Assets — The starting asset position for the NSN before any buys are made by ISAAC. The value in this field will only impact model output if the **Include Starting Assets?** options of the Parameters window is set to **Use Assets - InitAsset + FreeAsset**. In that case, the value in this field forces the model to set the item's target at least as large as the initial asset value and forces the resulting availability to reflect those assets. But those assets are not counted in the item buy total and hence are not included in the overall budget estimate. Field type: Text box; Field width: 7 numeric spaces.

NOP Flag — Enables the user to specify non-optimal spares options such as specifying a minimum spares target or a buy total. Processing of this field is controlled by the **Use Pre-specified Buy Quantity** parameter of the Stock / Resupply / Other Options Screen — described in Chapter 3 (Table 3-2) and used in conjunction with the Negotiated Level field quantity described next. The NOP Flag specifies the type of prespecified buy and the Negotiated Level field specifies the spares quantity. The following describes the field options - **NOP**, **FIX, ORD** or **AAA** (Field width: 3 character spaces).

– A value of 'NOP' will cause the model to develop a spares target of exactly the quantity of the component specified in Negotiated Level when the **Use Pre-specified Buy Quantity** is set to **Yes**. In this case, components with an entry of 'NOP' will be included in the total budget but will not be included in the availability calculation. When the **Use Pre-specified Buy Quantity** is set to **No**, the model will treat the quantity specified in Negotiated Level field as the target ceiling for the respective component.

– A value of 'FIX' will cause the model to develop a spares target of exactly the quantity specified in the Negotiated Level field and will include the component in both the budget and availability calculations regardless of the **Use Pre-specified Buy Quantity** field value.

– A value of 'ORD' will cause the model to buy at least the quantity specified in Negotiated Level. Thus, the model treats the quantity as a minimum value already on order and will be delivered prior to day "0" of analysis when the **Use Pre-specified Buy Quantity** field value is **Yes**. When the **Use Pre-specified Buy Quantity** field value is **No** the model will ignore the component quantity specified in the respective Negotiated Level field.

– Any other value (e.g., 'AAA' or MIN) will cause the model to treat the quantity specified in Negotiated Level as the floor for the respective component when the **Use Pre-specified Buy Quantity** is set to **Yes**. When the **Use Pre-specified Buy Quantity** is set to **No** the model will ignore the component quantity specified in the respective Negotiated Level field.

Negotiated Level — Negotiated level for this NSN or spares quantity of the component as determined by the NOP field value. Field type: Text box; Field width: 6 numeric spaces.

Item Buy — Percentage of the pipeline that is to be bought sacrosanct for this item. Applicable only when 'ITEM' is indicated in the respective **Force Buy based on Pipeline %** field on the Pipeline, Resupply & Options window (The **Force Buy based on Pipeline %** entered on the Stock Options section of the Pipeline, Resupply & Options windowas LRU percentage on first day, SRU percentage on first day, etc.) Field type: Text box; Field width: 5 numeric spaces (2 decimal places).

Conditional (Wartime / Non-wartime) Fields

These fields vary on the basis of war or nonwar conditions. These field values are affected by operational fields such as flying hours and / or maintenance fields such as failure rate. Each field has separate war and nonwar values. These fields are shown in Figure 4-22 and defined below.

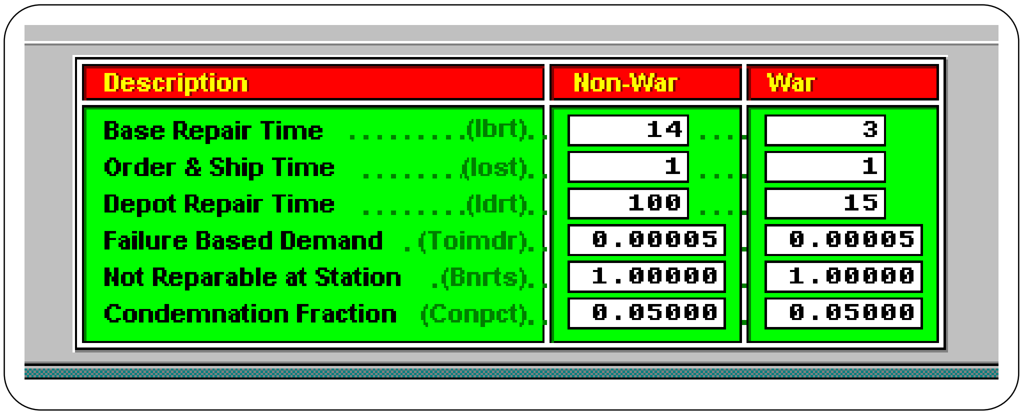


Figure 4-22.
The Wartime/Non-wartime Conditions Fields Section of the Kit Component Data Screen

Valid entries for each of the following fields are depicted in Table 4-2.

Base Repair Time — Base repair time (IBRT) in days for this component. This is the number of days from item failure through the repair of the item to a serviceable status. Field type: Text box; Field width: 6 numeric spaces.

Order & Ship Time — Order and ship time (IOST) in days for this component. This is the number of days from when a request is made on the depot for an item till that item is received in base supply. This does not include depot shortage time when a serviceable item is not available at the depot. Field type: Text box; Field width: 6 numeric spaces.

Depot Repair Time — Depot repair time (IDRT) in days for this component. This is the number of days of the complete depot repair cycle from NRTS of the item from the base to the depot through repair of the item at the depot. This includes the retrograde ship time from the base to the depot; but, it does not include the shipping time from repair until the item is received at the base (order and ship time). Field type: Text box; Field width: 6 numeric spaces.

Failure Based Demand — Wartime demand per flying hour for this component. The demand is based on the expected number of failures for this item. An item is classified as failing if it can only be recovered through procurement or repair. Field type: Text box; Field width: 8 numeric spaces (5 decimal places).

Not Reparable at Station — The NRTS (not reparable at this station) rate is an estimate of the percentage of the base demand that is not repaired at the base (that is the percentage that is sent to the depot for repair or condemned). To estimate the NRTS rate, you must split base demand into 3 component forecasts: 1) what percentage of the total base demand is repaired at the base, 2) what percentage of the base demand is repaired at the depot, and 3) what percentage of the base demand is condemned at the base or depot. The sum of those three equals 100%. Thus, the NRTS rate equals 100 minus the base repair percentage or the NRTS rate equals the condemnation percentage plus the depot repair percentage. If the item is only repaired at the base (i.e., there is no depot repair of the item) then the NRTS rate equals the condemnation percentage. Field type: Text box; Field width: 8 numeric spaces (5 decimal places).

Condemnation Fraction — Condemnation fraction for this component. This is the fraction of base demand that can not be repaired and must be condemned. Field type: Text box; Field width: 8 numeric spaces (5 decimal places).

Viewing Different Components via the Component List

You can select the item record you want to view by highlighting (click on) its NSN in the Component List. Component list information consists of two of the item identification data fields. These fields are shown in Figure 4-23 and their definitions are repeated below.

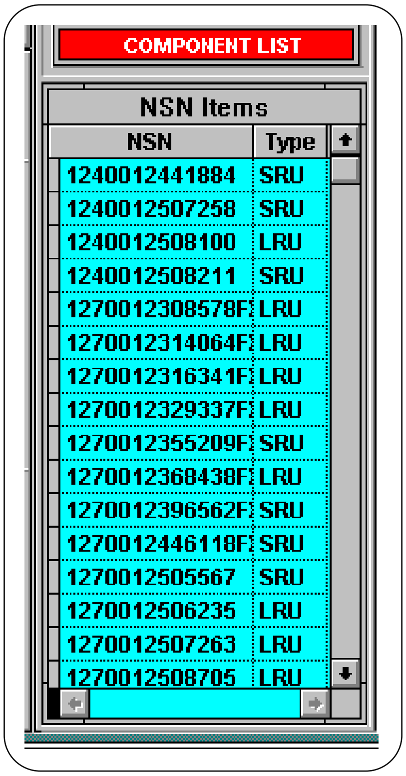


Figure 4-23.
The Component List Section of the Kit Component Data Screen

NSN — National stock number of the component.

Type — The type of reparable this spare is classified as (e.g., **LRU** or **SRU**).

How to Create New or Modify Existing Component Information

You can create new component records and modify and delete existing component records using the component screen. The top row of buttons of the Kit Component Data Screen (see Figure 4-14) is displayed alone in Figure 4-24. Those buttons are used for creating new component records and modifying existing item records. The buttons are briefly defined below and then we describe how to use those buttons in combination to edit component data:

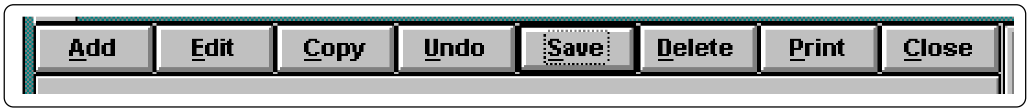


Figure 4-24.
The Top Row of Buttons on the Kit Component Data Screen

Add — Creates a new component record using the default component fields.

Edit — Enables you to edit existing component records.

Copy — Copies the highlighted record information to create a new component record.

Undo — Undoes the previous command.

Save — Saves the component field information.

Delete — This will delete one highlighted record at a time.

Print — This will provide access to a print menu. (Not currently working).

Close — Closes this screen and starts the indenture checking process. At the conclusion of the indenture checking process you will be returned to the Kit Parameters Screen.

Creating New Component Records

There are two options for creating new component records. One adds the default component fields and the other copies component fields from an existing component record in the current kit. In both cases you can only create one new component record at a time. Both options allow you to modify any or all of the component level information for a new component record. **Copy** is used instead of **Add** when the new record has more in common with a previous record than the item default record used in **Add**. We recommend that you choose the option that requires the least number of keystrokes.

Adding New Component Records Using the Default Fields

To add a new record (with the default fields) use the following sequence:

1. Click on the **Add** button.

2. Enter the National Stock Number (the field is already highlighted) and press enter.

3. Enter the Next Higher Assembly (NHA) identification number in the Next Higher Assembly field. For LRU component records, the NHA is automatically the weapon system name the user entered while defining the kit. For SRUs, the number entered must match one of the LRU National Stock Numbers in the Component List or one of the other SRU National Stock Numbers in the case of a lower indenture level SRU.

4. Change other component field information as required for this component. This is the only time you can change component field information without going into the edit mode.

5. Click on the **Save** button.

To create another new record, follow steps 1 through 5. To see the new component NSNs in the Component List it is necessary for you to click in the Component List (only after you have completed step 5).

Copying Existing Component Field Information into a New Component Record

To copy an existing record's component fields into a new record use the following sequence:

1. Select the record you want to copy by highlighting the NSN in the Component List.

2. Click on the **Copy** button.

3. Enter the National Stock Number (the field is already highlighted) and press enter.

4. Enter the Next Higher Assembly (NHA) identification number in the Next Higher Assembly field. For LRU component records, the NHA is automatically the weapon system name the user entered while defining the kit. For SRUs, the number entered must match one of the LRU National Stock Numbers in the Component List or one of the other SRU National Stock Numbers in the case of a lower indenture level SRU.

5. Change other component field information as required for this component. This is the only time you can change component field information without going into the edit mode.

6. Click on the **Save** button.

To create another new record, follow steps 1 through 6. To see the new component NSNs in the Component List it is necessary for you to click in the Component List (only after you have completed step 6).

Editing Existing Component Records

You can only modify one existing component record at a time.

1. Select the record you want to edit by highlighting the NSN in the Component List.

2. Click on the **Edit** button.

3. Change component field information as required for this component. This is the only time you can change component field information without going into the edit mode.

4. Click on the **Save** button.

Deleting Existing Component Records

You can only delete one record at a time using the **Delete** button. If you want to delete several records at one time, without deleting all of the records, you can tag multiple records for deletion as we will now describe.

Deleting One Record at a Time

To delete a record, highlight the NSN in the Component List (right side of the screen) and choose the **Delete** button. The record will not be deleted from view in the Component List until you click on the component list again. If you click on another field you will see the record you had selected for deletion "tagged" (a black box) in the delete column to the left of the NSNs.

Deleting Multiple Component Records

To delete several component records, "tag" each of them by clicking in the delete column in the Component List (to the left of each respective NSN). The records will not be deleted from view in the Component List until you click on the component list again. If you click on another field you will see the records you had selected for deletion “tagged” in the delete column to the left of the respective NSNs.

Warnings that Occur During Component Record Editing or Creation

The most likely warning to encounter is an “Edit Interrupt Error.” This will occur during **Add**ing, **Copy**ing or **Edit**ing a component record if you move your cursor to another record in the Component List before you have saved your changes to that component record. If you move your cursor Figure 4-25 will be displayed.

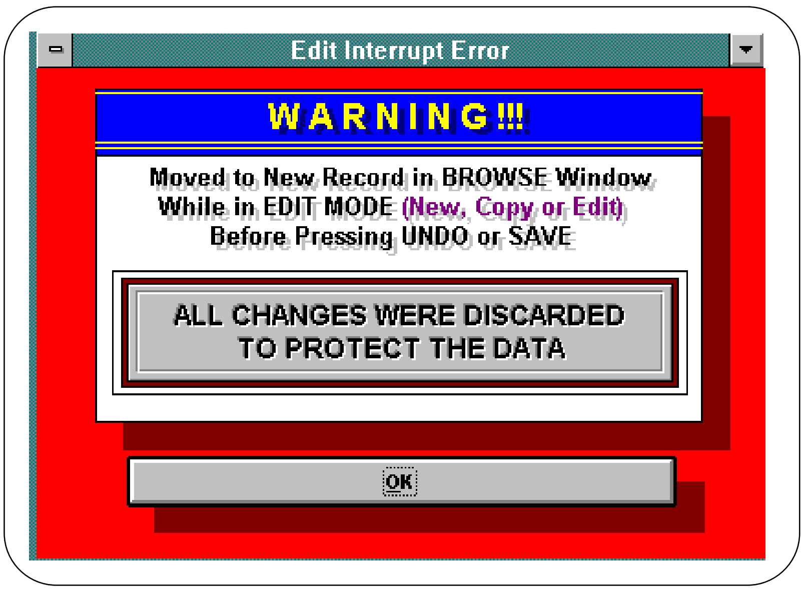


Figure 4-25.
The Edit Interrupt Warning

When the “Edit Interrupt Error” warning window appears, you will lose any changes you have made to field data since the last time you clicked the **Save** button.

You must close this window before further editing or model processing. You can close this window by either pressing the **‘Esc’** key or clicking on the menu control box.

Another type of warning that may occur is the LRU warning. This warning (Figure 4-26) will be displayed if you change the Rtype of a component record from SRU to LRU.

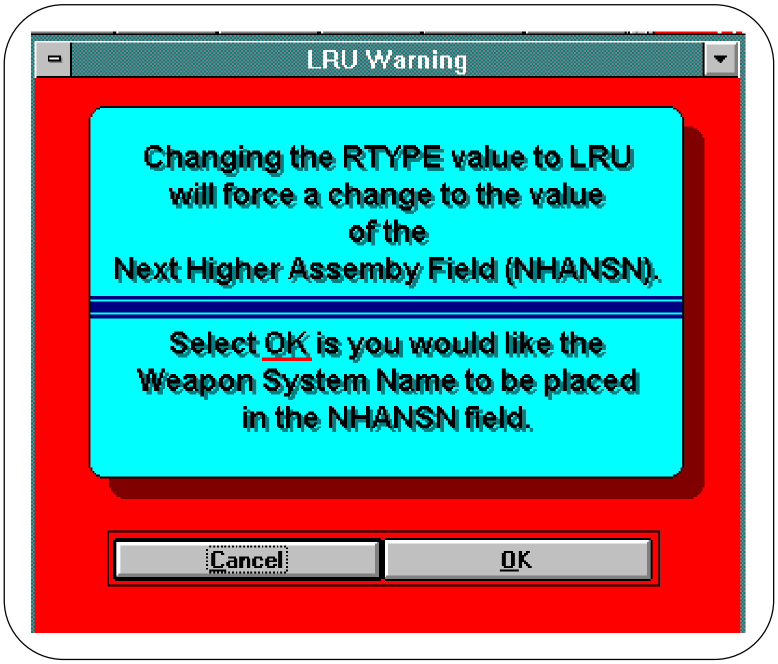


Figure 4-26.
LRU Warning

You have two options:

1. If you choose **Cancel**, ISAAC will cancel the change from SRU to LRU.

2. If you choose **OK**, ISAAC will accept the change from SRU to LRU and will change the Next Higher Assembly field value to match the value entered in the Weapon System field when the kit was defined.

Warnings Generated by Invalid Field Values

During the process of creating new or editing existing kits you may see one or more errors or warnings displayed on your screen. Warning windows notify you that there is a non-catastrophic problem with your data, your key strokes or your mouse movement. The error that caused the warning is a non-fatal error. The changes that you made will either be discarded or corrected with valid entries so as to protect the database from being corrupted with invalid data.

Validation for Component Fields

Table 4-2 lists each of the component data fields in conjunction with each field's valid range of values. If you enter a field value that is not within the valid range, the model displays a warning message and allows you to reenter the data.

Table 4-2.
Valid Range of Values for Component Data Fields

Field Name	Type	Width	Decimals	Valid Range of Values
National Stock Number	Character	18	0	Not blank
Type (Drop Down Box)	Character	4	0	LRU or SRU
Next Higher Assembly	Character	18	0	Not blank
Cost	Numeric	11	2	Must be greater than 0
MAINTCON (Drop Down Box)	Character	3	0	RR No War Repair orRRR War Repair
Quantity on Parent	Numeric	4	0	Must be greater than 0
PLTT	Numeric	6	1	Must be greater than 0
Budget Code	Numeric	2	0	Whole numbers from 1 to 99 inclusive
Early NRTS (Drop Down Box)	Numeric	5	2	1 = Standard or 2 = Exception
Future Application	Numeric	6	2	Real numbers from 0.01 to 1.00
Cannibalization Flag(Drop Down Box)	Character	1	0	Y or N
Item Name	Character	19	0	Not blank
NOP Flag	Character	3	0	Not blank
Negotiated Level	Numeric	6	0	Must be greater than 0
Initial Assets	Numeric			
Item Buy	Numeric	5	2	Real numbers from 0.00 to 1.00
Manager	Character	4	0	Not blank
Base Repair Time	Numeric	6	0	Must be greater than or equal to 0
Order and Ship Time	Numeric	6	0	Must be greater than or equal to 0
Depot Repair Time	Numeric	6	0	Must be greater than or equal to 0
Failure Based Demand	Numeric	8	5	Must be greater than or equal to 0.00000
Not Reparable at Station	Numeric	8	5	Real number from 0 to 1 inclusive
Condemnation Fraction	Numeric	8	5	Real number from 0 to 1 inclusive and <= NRTS

Developing the Component Indenture Structure

Once all component information has been entered and the user clicks the close button, the model develops the indenture structure. The indenture structure describes the relationship between components. The aircraft is composed of LRUs (1st level of indenture), LRUs are composed of specific SRUs (2nd level of indenture), and those SRUs are composed of lower indenture SRUs (3rd level of indenture), and so on. The key field the model uses to develop the indenture structure is the NHANSN because it specifies the parent of a particular NSN.

This structure is similar to a family tree with the highest level (the LRUs on level 1) having the greatest impact on aircraft availability. A more-expensive LRU has a greater impact on availability (when a spare LRU is available, the aircraft is operational almost immediately), while the usually less-expensive SRU affects availability only indirectly (even if an SRU spare is immediately available, the aircraft must wait for the LRU to return from maintenance).

To develop the indenture structure, the model starts processing at the LRU level and determines which children (2nd indenture level) belong to them (i.e., which children have an NHANSN that equals one of the LRU NSN values). Next, the 2nd level indenture items are treated as the parents and the model determines which children belong to them. This process repeats itself until the model finds no more children.

Common SRU Components

Specifying indenture for most components is relatively simple, the user specifies a single record for each item with a unique NSN and NHANSN. However, some SRUs are common to more than one item within a particular aircraft. In that case, the user may need more than one record for a common SRU. Figure 4-27 represents a simple aircraft with two types of common SRUs:

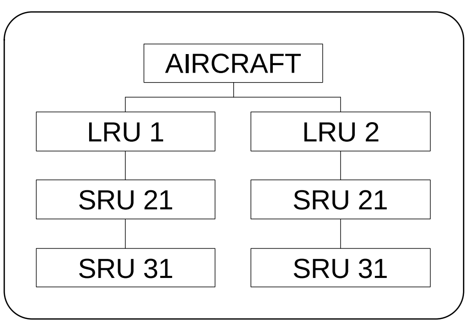


Figure 4-27.
Aircraft with Common SRUs

– A common SRU on two different parents. That is the case for SRU 21 which is on 2 different LRUs (1 and 2). The user needs to represent that indenture structure with 2 different records since there are 2 different parents.

– A common SRU on the same parent but on 2 different LRUs. That is the case for SRU 32. The user needs to represent this structure with 1 record only since there is only one parent and ISAAC automatically generates the common SRU occurrences.

Kit Input Information

Type	NSN	NHANSN
LRU	1	Aircraft
LRU	2	Aircraft
SRU	21	1
SRU	21	2
SRU	31	21

Indenture

Model Component Data

Type	NSN	NHANSN	LRU NSN
LRU	1	Aircraft	
LRU	2	Aircraft	
SRU	21	1	1
SRU	21	2	2
SRU	31	21	1
SRU	31	21	2

Figure 4-28.
Difference Between Kit and Model Component Indenture Structure

Figure 4-28 displays the user kit information required to develop the indenture tree displayed in Figure 4-27. Once the user enters the information, the model indenture processing automatically generates the full indenture structure with each of the common SRU occurrences (e.g., adding the multiple occurrences of SRU 32). ISAAC generates the full indenture structure to reduce the data input requirements of the user and simplified his task. *A key point to remember is the common SRUs with different parents (e.g. SRU 21) kit information must only differ for the NHANSN field andQPANHA and all other fields must be identical.*

A final indenture note, ISAAC displays component information in three basic aggregations: (1) the kit input information that we have just described contains all records where the NSN plus NHA fields are unique (left side of Figure 4-27); (2) the model input component information contains all records where the NSN, NHA, and LRU NSN fields are unique (right side of Figure 4-28 and in the **View Input** Component data browses); and (3) the model output information contains all records where the NSN fields are unique (in the**View Output** browses).

Errors and Warnings from Creating the Indenture Levels

By developing this indenture structure, ISAAC can identify data problems that would cause an invalid indenture structure and wrong model results. When a type of problem is discovered, the model displays it as either an “Error:” (allows the user to correct it) or as a “Warning:” (the model automatically corrects the information without user assistance) window. The model identifies 7 types of problems:

1. *Error - Duplicate NSN+NHANSN Record.* This error indicates that two or more records have the same NSN and NHANSN values in their field or a duplicate record. The model will display the extra records.

2. *Error - NSN=NHANSN.* This error indicates that a record has the same value for the NSN and the NHANSN.

3. *Error - NSN or NHANSN BLANK.* This error indicates that the user did not enter any value for the NSN or the NHANSN fields.

4. *Error: No valid NHANSN parent found.* This error indicates that the item’s NHANSN does not match any of the NSN numbers in the indenture structure.

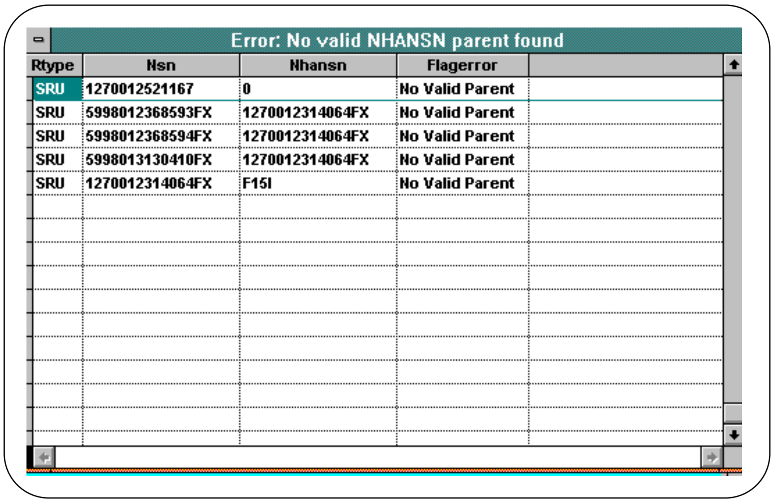
5. *Error: Indenture level greater than maximum level of 5.* Though ISAAC initially develops an indenture structure with as many as 10 levels, it deletes all SRUs on any level greater than 5, assuming the that information is erroneous. If your database has more than 5 levels the model can be easily adjusted.

6. *Warning- Item on 2 Levels.* This error indicates that an item is on two or more different indenture levels. The model will automatically delete the item with the highest indenture number (2, 3, etc.).

7. *Warning: Quantity/plane not equal (will overwrite with ISAAC).* This error indicates that the imported database contains the IQPA field. That field represents the total number of applications on the aircraft. The model calculates this field value while developing the indenture structure (by summing the number of applications on each of the component’s next higher assemblies). As a check, ISAAC compares the calculated value with the imported value to help identify inconsistencies. Since ISAAC automatically overrides the imported data with the calculated data, no user action is required (though after processing you may want to check what caused the mismatch).

In creating that structure, the model checks the indenture data in a specific sequence to makes sure it is valid. The model will conduct no further checks or processing while an “Error:” or “Warning:” window is open. You must close this window before further model processing. You can close this window by editing the data (such as a NSN or NHANSN number, move off the edited field to see if the edit was entered correctly, and then click on the menu control box to end the edits.

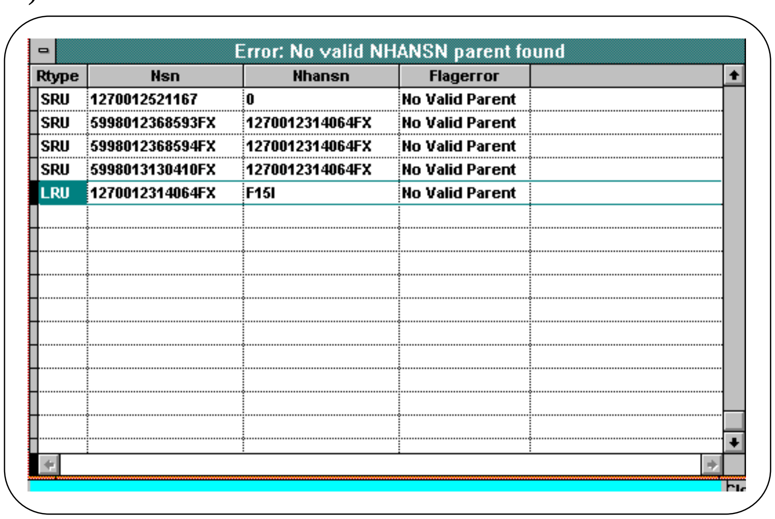
For instance, one type of error occurs when there is no valid Next Higher Assembly found in one or more component records. This results in a “Please Edit - Error: No valid NHANSN parent found” window (Figure 4-29). This is a fatal error that you should correct when it appears. Your corrections are made on the error window itself.


Error: No valid NHANSN parent found

Rtype	Nsn	Nhansn	Flagerror
SRU	1270012521167	0	No Valid Parent
SRU	5998012368593FX	1270012314064FX	No Valid Parent
SRU	5998012368594FX	1270012314064FX	No Valid Parent
SRU	5998013130410FX	1270012314064FX	No Valid Parent
SRU	1270012314064FX	F15I	No Valid Parent

Figure 4-29.
Error: No Valid NHANSN Parent Found

In Figure 4-29 we changed the Rtype for the fifth record from SRU to LRU. In this example, this will correct the problems with records 2 through 5 displayed in this window. We made no changes to the first record (it still has an error).


Error: No valid NHANSN parent found

Rtype	Nsn	Nhansn	Flagerror
SRU	1270012521167	0	No Valid Parent
SRU	5998012368593FX	1270012314064FX	No Valid Parent
SRU	5998012368594FX	1270012314064FX	No Valid Parent
SRU	5998013130410FX	1270012314064FX	No Valid Parent
LRU	1270012314064FX	F15I	No Valid Parent

Figure 4-30.
Error Window with Correction Highlighted

The model will complete the rest of the indenture checks without using your corrections.

Once the model has conducted all of the indenture validation checks, you will be given an opportunity to rerun the indenture process with your corrections. The mechanism for correction is the “Error Alert” window (Figure 4-31).

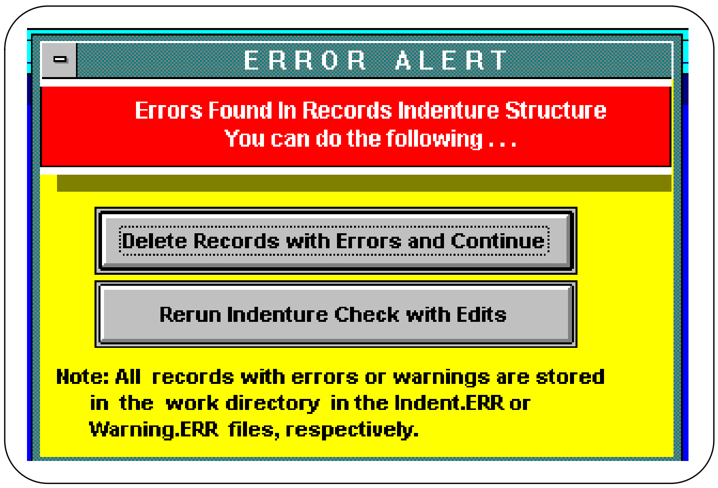


Figure 4-31.
Error Alert Window

At this point, you have two options.

– You can delete all component data records with fatal errors in them. This is accomplished by choosing **Delete Records with Errors and Continue**. In the example, the records with invalid NHA’s would be deleted (ignoring your edits); while records with mismatched QPAs (i.e. QPA field value doesn't match model calculated QPA) would have the QPA changed to match the model calculated QPA. This option returns you to the Kit Parameters Screen.

– You can rerun the indenture process including the corrections you made on the error windows. This process is initiated by choosing **Rerun Indenture Check with Edits**.

- ► This option reruns the indenture checks. Any uncorrected errors or additional errors caused by your corrections will appear on the appropriate error window (if applicable).

- ► If you have uncorrected or additional errors this indenture check process will be repeated. Make your corrections on the appropriate error window.

- ► The indenture check process will continue until all fatal errors have been corrected or the records with the errors have been deleted.

GLOBAL CHANGES (SENSITIVITY) TO KIT

If you select **Global Changes (Sensitivity) to Kit** from the **Kit** pull-down menu, ISAAC lets you select an existing baseline kit, make a copy of it and then make changes to multiple records at a time. The copied kit will have the same global parameters as the kit it is copied from. The component records will initially be exact duplicates of the component records of the copied kit. You can then make global changes to all records (e.g., increase all base repair times by 10%) or make changes to records that meet user specified filter conditions (e.g., increase only those base repairs times for items with costs greater than $200,000).

This section is organized in four parts. The first part summarizes the basic sensitivity change sequence. The second part walks you through a basic (global) sensitivity change sequence. The third part presents the more complex sensitivity change sequence that employs user-specified filters to determine which records to change. The last part is a miscellaneous section.

Basic Global Changes Sequence

1. Select **Global Changes (Sensitivity) to Kit** from the **Kit** pull-down menu.

2. Choose a baseline kit from the Available Baseline Kits window.

3. Define Kit ID: Enter **Kit Name**, unique **Description** and **Weapon System**.

4. Sensitivity Change Page Frame is displayed.

5. Define Changes page of the Sensitivity Change Page Frame is displayed initially. Select the field that you want to change. Pick the type of change to apply from the **Change Type** drop-down list box. Enter the amount of change in the **Change Amount** text box. It is important that you enter a value in the **Change Amount** text box, even if the default value displayed is the value you want to apply. Once you are done select the **Calculate** button.

6. Preview / Edit / Apply Changes page then appears. Preview and edit field values if necessary. Once you are satisfied **Apply** the changes.

7. Repeat steps 5 and 6 for each field that you want to change. Thus, ISAAC allows you to make multiple global changes to the fields of the kit by repeating steps 5 and 6.

8. Select **Save & Exit** from any page of the Sensitivity Change page frame.

9. Field Default Values Screen is displayed. Make any changes to the default values if applicable and select **Close** when you are done. (From this point of the process on it is similar to the process of importing a kit.)

10. Checking for Field Values Out of Range window is displayed while the model checks for out-of-range values that you might have created during the sensitivity session.

11. Kit Parameters Screen is displayed once error checking and replacement is finished.

Detailed Global Changes

Select **Global Changes (Sensitivity) to Kit** from the **Kit** pull-down menu, ISAAC displays the Available Baseline Kits window (Figure 4-2). Move the cursor to the kit you would like to copy and then select it by pressing **'Enter'** or by clicking the right mouse button. ISAAC displays the Define Kit ID window (Figure 4-3).

You must enter a **Kit Name**, **Description** and **Weapon System** for the new kit in the respective text boxes. Your (kit) **Description** must be unique and should be meaningful to you so as to aid in future identification of this kit. The **Weapon System** is used in the model to identify the kit information. Once you have entered the information select **OK**.

Soon the Define Changes page of the Sensitivity Change page frame will appear (Figure 4-32).

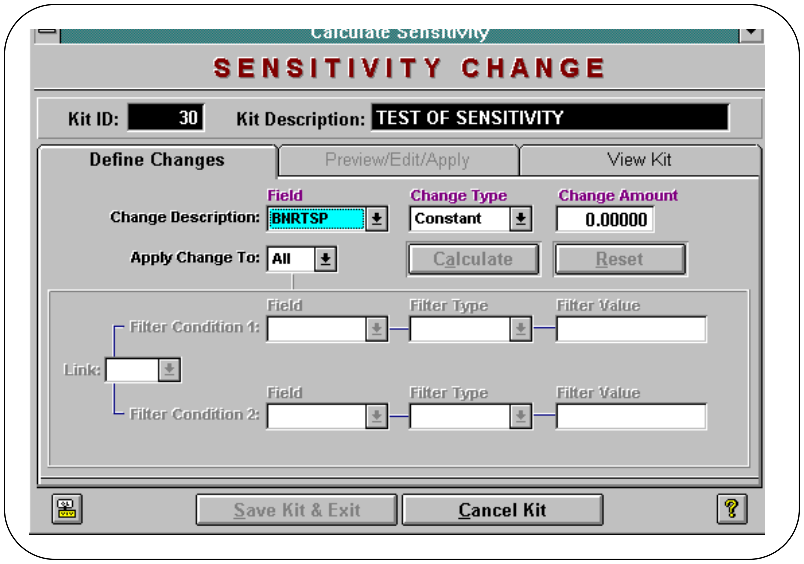


Figure 4-32.
Define Changes Page of the Sensitivity Change Page Frame

At this point you can define the changes that you want to make to the component records. ISAAC allows you to make one or more global changes (although, you can only change one field at a time), to make changes on the basis of data meeting user specified filter characteristics, or a combination. We will limit our change to a single global change for the time being. We will change the wartime demand rate (TOIMDRW) to be 50% of the original value (see Figure 4-33).

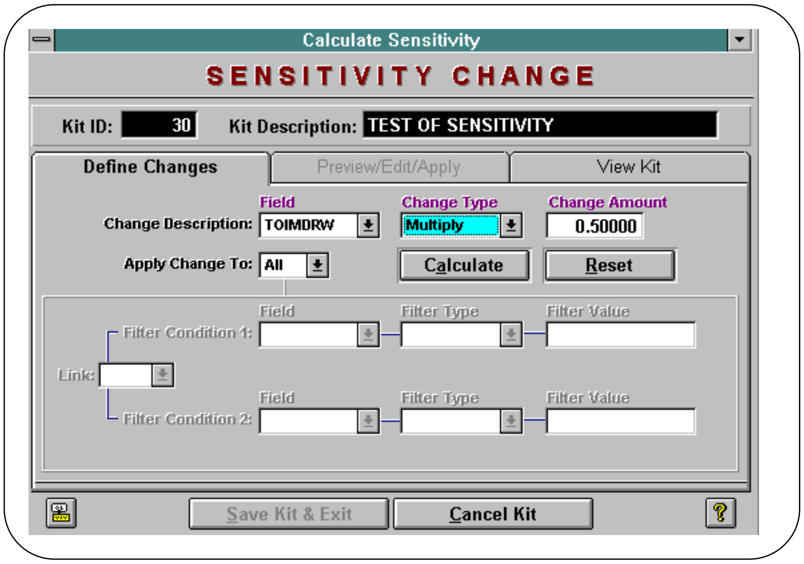


Figure 4-33.
Sample Global Change — Multiply War Demand Rate by 0.5

We select the field and type of change and enter in the amount of change to the **Change Amount** text box on the Change Description line. In our example we select the TOIMDRW field from the **Field** drop-down list box. There are two methods to select a field from the **Field** drop-down list box. You can scroll to the field using the vertical scroll bar or you can type in the first letter of the field name to bring you closer to the field that you want to select. We then choose **Multiply** from the **Change Type** drop-down list box and then we enter **0.5** in the **Change Amount** text box. <u>It is important that you enter a value in the **Change Amount** text box, even if the default value displayed is the value you want to apply.</u>

We then determine how to apply the change. We have two choices: we can apply the change to all records or we can set filters to limit the changes to certain records. In our simple example we will apply the change to all records by selecting **All** from the **Apply Change To:** drop-down list box.

Up to this point we have defined the change(s) but not calculated or applied the change(s) to the kit component records. We now select the **Calculate** button on the **Apply Change To:** line. At this point the Preview/Edit/Apply page of the Sensitivity Change page frame (Figure 4-34) will be displayed. You must enter an amount or value in the **Change Amount** text box to activate the **Calculate** button.

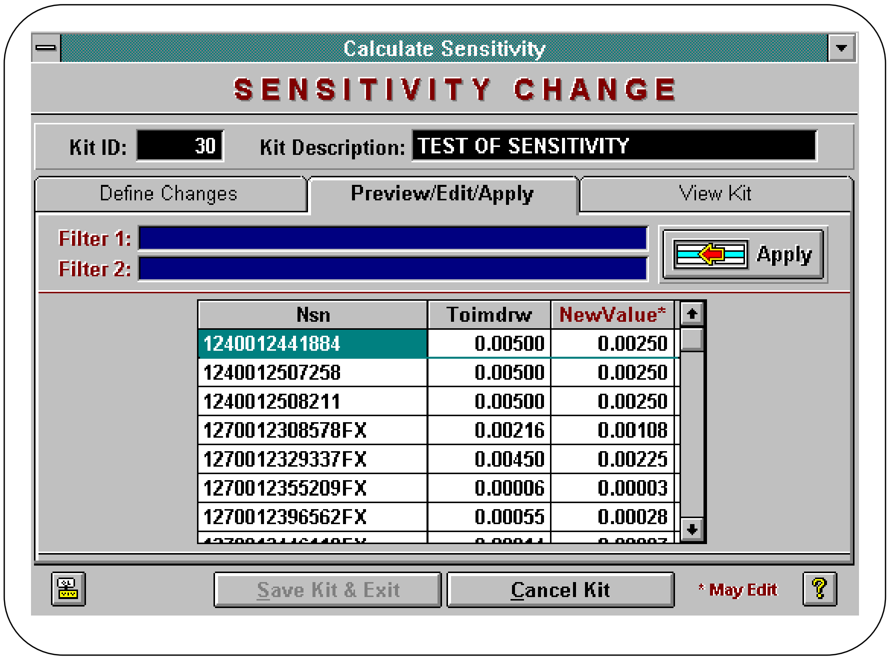


Figure 4-34.
Sample Global Change — Preview/Edit/Apply Multiply War Demand Rate by 0.5

This page will display the New Value (heading typed in red letters) for each record next to the original value located in the column with the respective field name at the top (in this case TOIMDRW). The records are listed in NSN order from lowest to highest numerically. At this point, you can edit one or more of the values in the New Value column to handle special cases. You have two choices at this point. If you select the **Apply** button it will apply whatever value is in the New Value column to the specified record, or you can exit out of the sensitivity change mode without making the changes altogether by selecting the **Cancel Kit** button.

Notice, in the current example some of the New Values are not 50% of the old values. ISAAC rounds up to the last decimal point (e.g., ISAAC calculates 0.5 * .000005 as .000003 versus .0000025). This is also true when dealing with integer values such as the repair times where ISAAC again rounds to the nearest integer.

Also notice that the Sensitivity Change page frame contains several pages. These pages are tabbed like a file folder and each page has the functionality of a window. You can move back and forth between the pages by clicking the tabs at the top of each respective page. Thus, you can return to the define changes folder to readjust your sensitivity changes or make additional changes to other fields. If you move to another page before **Apply**ing the changes you will lose the changes (i.e., the data will not be changed). You can also move to the View Kit page at any time to help you remember just what values the kit already contains or to view all kit fields to view applied changes.

You can make as many data changes to as many fields as you would like by repeating the steps just described. However, you can only change one field per record with each sensitivity change. You can even make multiple changes to the same field more than once. After you have applied your change(s) you can view the changed field data via the View Kit page of the Sensitivity Change page frame. Once you have applied all of the changes that you want to make, you can select the **Save Kit & Exit** button when you have completed your changes.

At this point, ISAAC checks to ensure all changes are within the specified valid field ranges (see Table 4-2). You now see a window, that looks similar to Figure 4-10, that allows you to change the item level default values used when ISAAC identifies an out-of-range field (as described earlier). Make any changes that you want to the Field Default Values Screen and then select **Close**. The Checking for Field Values Out of Range window is displayed. If there are any out-of-range values found, the Warning: Found Field Errors, Errors Replaced by Default values window is displayed. Click the **OK** button to close this window. The field.err window will be displayed. It will list any field errors and their replacement values along with the NSN(s) that identifies the records. Close this window by pressing the '**Esc**' key.

The Kit Parameters Screen will appear (Figure 4-14). The information displayed on the Kit Parameters Screen as well as all other windows that can be accessed from this screen (through the **Scenario Baseline**, **Stock / Resupply / Other Options**, and **Component Data** buttons) is an exact duplicate of the baseline kit that this was copied from. For more information on the Kit Parameters Screen see the Kit Parameters section.

Global Changes with Filters

The previous sensitivity change example applied to global changes to the data. ISAAC has the capability to make changes to a user-specified subset of records. ISAAC allows you to constrain the records that will be changed by setting up to two data filters.

We will now walk you through a two-filter example. In our example, we want to reduce the non-war base repair time for Remove and Replace (RR) components that cost more than $50,000 by 50%.

We proceed through all steps of the Global Changes (Sensitivity) to Kit sequence up to opening the Sensitivity Change page frame.

1. Select **IBRTP** from the **Field** drop-down list box.

2. Choose **Multiply** from the **Change Type to:** drop-down list box (division is not an option).

3. Type in **.5** in the **Change Amount** text box. <u>It is important that you enter a value in the **Change Amount** text box, in order to activate the **Calculate** button for use in step 8</u>.

4. **From the Apply Change To:** drop-down list box select **Filter**. This activates the Filter Condition 1 line and the **Link** drop-down list box.

5. Select **MAINTCON** from the **Field** drop-down list box; choose **Equal to** from the **Filter Type** drop-down list box; and type **RR** in the **Filter Value** text box.

6. Choose **And** from the **Link** drop-down list box. This activates the Filter 2 line.

7. Select **Cost** from the **Field** drop-down list box; choose **More Than** from the **Filter Type** drop-down list box; and type **50000** in the **Filter Value** text box. Your page should look like Figure 4-35 at this point.

8. Select the **Calculate** button and the Preview/Edit/Apply page will be displayed (similar to Figure 4-36).

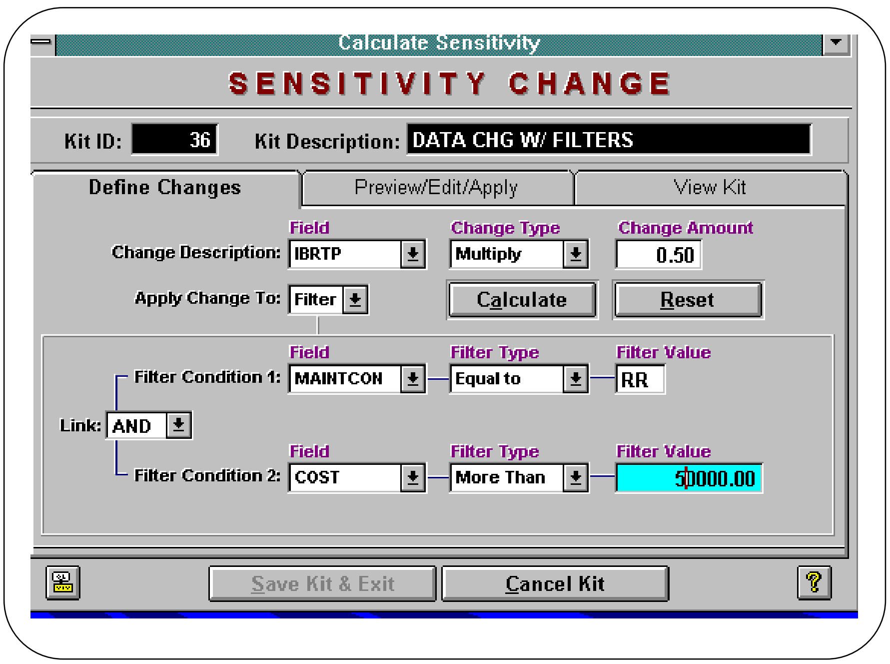


Figure 4-35.
Sensitivity Change Using Filters

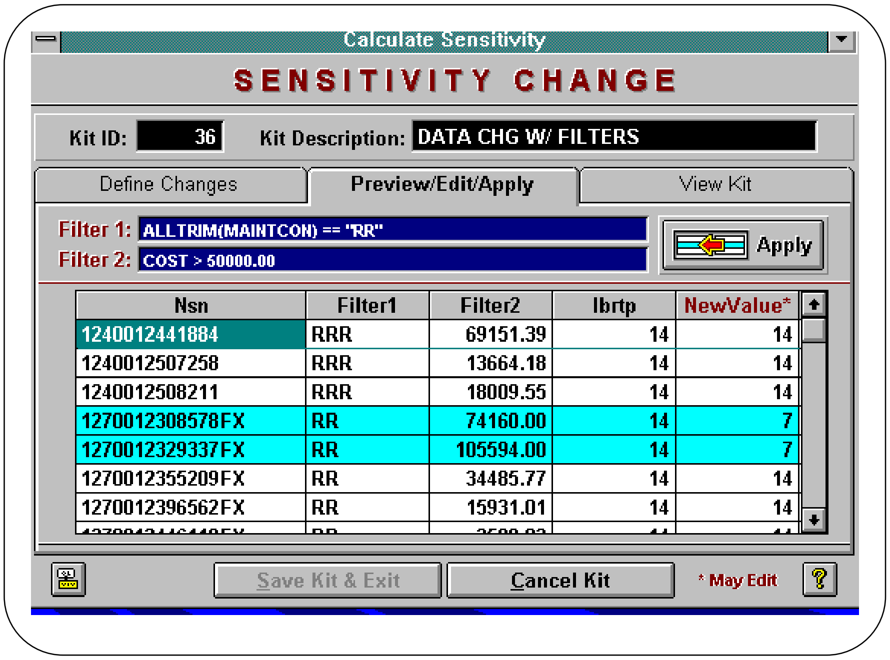


Figure 4-36.
Sensitivity Change Preview/Edit/Apply Page Using Filters

Notice that some records are highlighted in blue while others are not. Only those records that have met all of the filter criteria will be highlighted. Above the record data are two lines in dark blue labeled Filter 1 and Filter 2. These lines display the filter(s) that you have set. Once you are satisfied with the changes (and make any edits if necessary), select **Apply** to apply your changes.

You may then make other sensitivity changes applying the changes to **All** or **Filter**, as necessary. Before defining a new set of changes, you should press the **Reset** button on the Define Changes page. This button will reset all fields on the page to their default values and remove the old filters. It will not make any changes or reset any values in your kit component database. Once you are through making changes select **Save Kit & Exit**.

Miscellaneous

This section will define some terms unique to the Sensitivity Change page frame of ISAAC (see Figure 4-37). We will start with some of the general features that always apply to the pages. Then we will describe the Filter Type functions that apply to filtered sensitivity changes only.

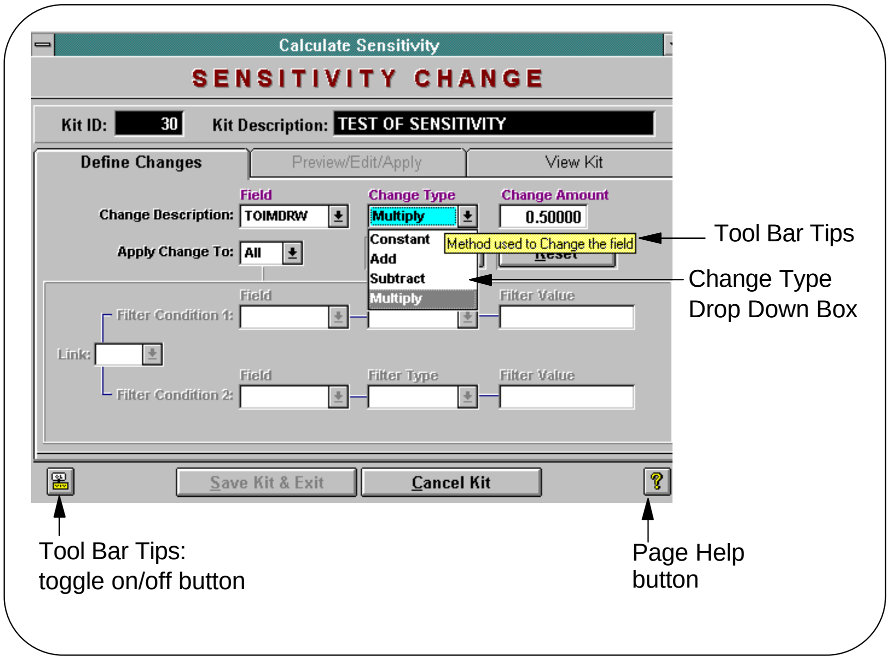


Figure 4-37.
Screen Features of the Sensitivity Change Screen that are New to ISAAC

Apply Change To: (Drop-Down List Box)— Allows you to either select **All** records to change or to use a **Filter** to limit the records to change.

Change Type (Drop-Down List Box) — There are two types of Change Type Drop down boxes. For numerical fields you have four options **Constant**, **Add**, **Subtract**, and **Multiply**. For character fields the only change is a **Constant** change.

Field (Drop-Down List Box) — This is a list of the component record fields that you can apply changes to. You can only make changes to one field at a time per sensitivity change.

Page Help Button — Displays a description of how to use the page that you are "opened" to.

Tool Bar Tips — Tips that describe the function of each drop down box and each of the buttons on each page of the Sensitivity Change page frame. When your mouse arrow rests on a button a tool tip will appear in a yellow rectangle.

Tool Bar Tips Button — Is a toggle button that turns on (tool tips will be displayed) and off (tool tips will not be displayed) the tool bar tips.

Field (Drop-Down List Box see Figure 4-38) [Filter Condition 1 and 2] — This is a list of the component record fields that you can use as a filter. You can select up to two fields per sensitivity change (one for each filter condition).

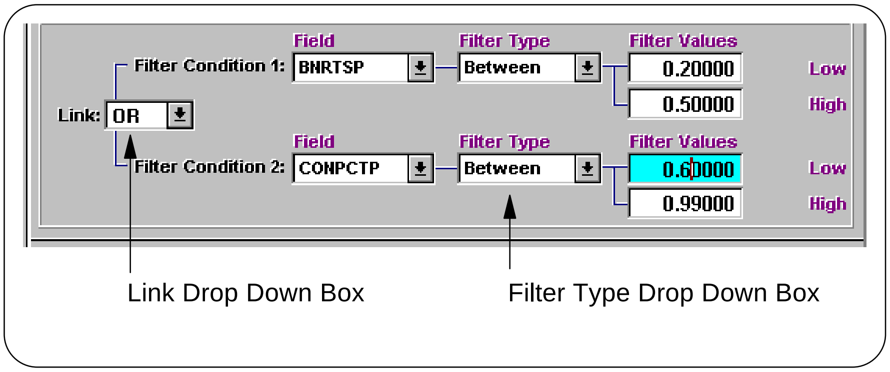


Figure 4-38.
Link and Filter Drop Down Boxes

Filter Type (Drop-Down List Box) [Filter Condition 1 and 2] — The logical Filter Type that you want to apply. The choices are **Equal To**, **Not Equal To**, **Less Than**, **More Than**, **Between**, and **Not Between** for numeric fields and **Equal To,** and **Not Equal to** for character fields.

Filter Value (Drop-Down List Box) [Filter Condition 1 and 2] — The filter value that you want to apply. There is generally one filter value per filter condition; except, when the Filter Type is **Between** or **Not Between**. In the latter cases you must enter two separate filter values into two separate boxes. The lowest numerical value should be entered in the upper text box.

Link (Drop-Down List Box) — For two filters only. You can choose **AND** to apply both filter conditions simultaneously or choose **OR** to select records that meet one or both of the filter conditions.

Chapter 5

Input, Output and Miscellaneous Data Windows

This chapter presents a detailed description of the reports (which we refer to as windows) that display the input and output data of ISAAC. This includes detailed item information such as demand rates, repair times, and spare buy totals as well more general information such as total aircraft performance or total spare budgets. We have divided this chapter into the following six sections.

- Browse — This section provides information on the **Browse** pull-down menu. This menu can change the appearance of the data displayed in the windows covered in the last three sections of this chapter.
- Printing Input and Output Field Data — This section describes how to print input and output field data using any of the picklists (the user mechanism used to select specific data fields to browse).
- Indexed Browse Windows — This section describes the general characteristics of and the sequence of operation for the group of windows that enable you to sort on multiple fields.
- Input Data Windows — This section consists of those windows where only input parameters can be viewed.
- Output Data Windows — This section consists of those windows where only output parameters can be viewed.
- Miscellaneous Data Windows — This section consists of those windows where combinations of input and output parameters or shopping lists from two separate runs can be viewed.

The first two sections apply to all of the input, output and miscellaneous data windows that are accessed through one of the pull-down menus and a browse picklist (these sections do not apply to the **Curve**, **Performance Report** and **View Stats** windows because they are not accessed via picklists). The third section applies to the data windows that can be sorted on multiple different fields. The last three sections describe each one of the data windows separately. We define each data field presented in the last three sections in Appendix B.

Browse

Browse allows you to examine and make changes to a database. In a browse window, the name of the database appears as the window title. The contents of the database appear in the window. The browse window is unique because you can split it into two partitions and examine different parts of your database at the same time. In addition, you can move, resize fields, and change views. [1]The ISAAC windows are modified browse windows because they provide a subset of the browse capabilities for the respective window. In general, modified browse windows will not allow you to change field data or to append or delete records. Specifically, only those browse capabilities that are highlighted on the **Browse** pull-down menu options will be accessible (see Figure 5-1).

Moving The Window

You can move a window by dragging the title bar of the window to the new location. As you move the mouse pointer, an outline of the window moves with it. Release the mouse button when the window is where you want it to be.

To cancel the move, you must press '**Esc**' before you release the mouse button.

Resizing The Window

To change the size of a window, point to a border or corner of the window. The pointer changes to a two-headed arrow. Drag the corner or border until the window is the size you want. If you drag a border, the window size changes only on the side of the border you drag. If you drag a corner, the two adjoining sides that form the corner change size at the same time. An outline shows the size and shape you selected.

Release the mouse button when the window is the shape you want it to be. To cancel the resizing, you must press '**Esc**' before you release the mouse button.[2]

Splitting The Window

You divide the browse window with a window splitter located in the lower left corner of the window. When the browse window is divided, the database appears in both partitions. For instance, the splitter lets you always see the NSN

---

[1] This paragraph is taken directly from FoxPro Interface Guide, Fox Software Inc., May 1991; pp. 5 – 12.

[2] “Moving the Window” and “Resizing the Window” are largely based on from Microsoft Windows User’s Guide (Version 3.1),1992; p. 13.

number in the left side of the browse while changing fields in the right side. To split the window into two partitions using the mouse, position the pointer on the window splitter (black box in the lower left corner - see Figure 5-1). Once the double headed arrow appears, drag the splitter to the right to open the left portion (or to make the left portion larger): drag the splitter to the left to make the left portion smaller (or to close it altogether)[3].

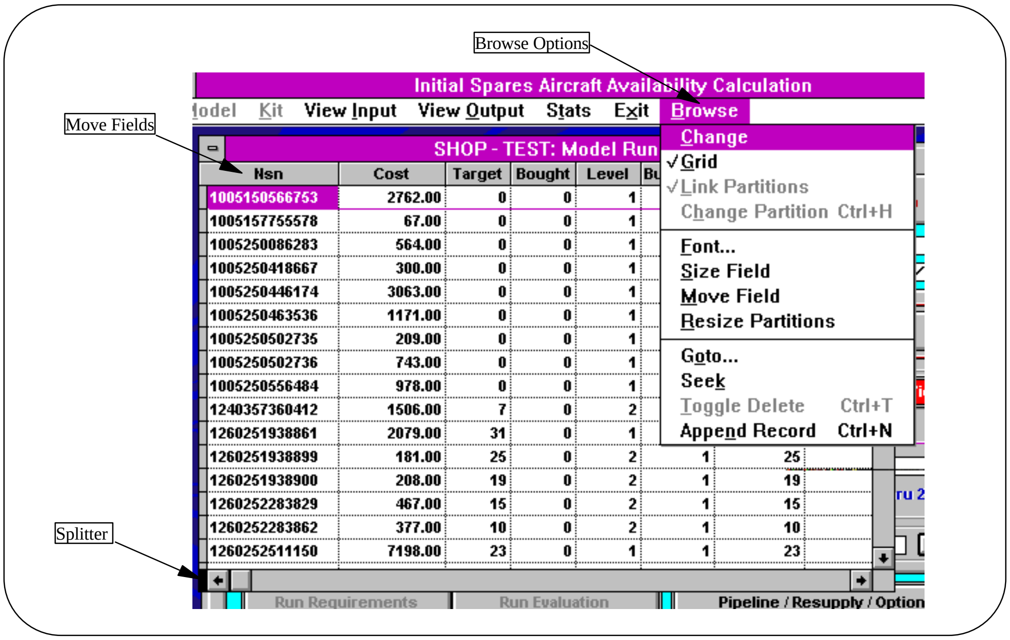


Figure 5-1
The Browse Pull-down Menu

The **Link Partition** and **Change Partition** menu options (described in the following section) are disabled until you partition the window. The two ISAAC windows that are accessed by the View Input pull-down menu are partitioned when you open them (Component Data and Run Log Windows). The other modified browse windows are not partitioned when you open them; however, they can be partitioned after you open them.

Changing Displayed Field Size

To change the display size of a field, move your cursor to the right border of the header of the appropriate field (the move field box of Figure 5-1 points to the gray header area) until your mouse arrow turns to a double headed arrow. Once the double headed arrow is available, hold down the mouse button and

[3] This paragraph is taken directly from FoxPro Interface Guide, Fox Software Inc., May 1991; p. 5-15.

drag the border to the appropriate size. When the field is the desired size, release the mouse button. You can resize as many fields as you would like, one at a time.

This sizing does not change the actual width of the fields in the database, only the display width. If some of your data appears to get cut off when sizing, don't worry. The actual underlying data is still there. When the window is split, field sizing affects both partitions.

Moving Fields

To change the sequence of the fields presented, with a mouse, press on the field header (for location see the *move fields box* in Figure 5-1) and move the field to the location you want. When the field is where you want it to be, release the mouse button. This capability allows you to change the order of the fields in the browse window. This does not change the actual order of the fields in the database, only the display order. When you move a field in one partition of a split window, the same field moves to the same location in the other partition.

Browse Pull-down Menu

The **Browse** menu contains options for use in the browse window. Some options are available only when the window is split[4]. Other options are only available for specific browse windows (i.e., the **Seek** option is only available on the browse windows that contain item level information and the Multi-Day Evaluation Window). The browse options make it easier for the user to find and view portions the data by allowing a number of different ways to view it. ISAAC browse windows do not let you change the data only the display of the data.

Change

Displays the database in the browse window in the Change mode. The record selected in the browse window will be displayed. In Change mode, the fields in each record are listed one below the other horizontally. If the window is split, only the active partition appears in the Change mode when you choose **Change**.

Change toggles between **Change** and **Browse** depending on the mode of the browse window.

---

[4] The next 11 sections (**Change** through **Append Record**) are taken directly from FoxPro Interface Guide, Fox Software Inc., May 1991; pp. 5-13 – 5-15. Only those sections that specifically deal with the ISAAC model or its specific windows are original prose.

Grid

Removes the vertical lines between field data in the browse window. If the window is split, only the active partition is affected.

The **Grid** option toggles between **Grid** (Off) and **Grid** (On) depending on the current grid setting.

Link Partitions

Displays a vertical scroll bar in both partitions so that you can scroll each partition independently of the other. This option is enabled only when the window is split. When only one scroll bar is displayed, the partitions are linked and will scroll together.

The **Link Partitions** menu selection is a toggle between linked partitions (when it is checked) and unlinked partitions (when the menu selection is not checked).

The **Component Data** and **Run Log** windows (from the **View Input** pull-down menu) are partitioned upon opening the respective window. These two windows have two vertical scroll bars that will not operate independently unless you toggle **Link Partitions** to unlink the partitions.

Change Partition

Makes the inactive partition of the split window active, and the active partition inactive. This option is enabled when the window is split.

Size Field

You can resize fields using the **Size Field** selection from the **Browse** pull-down menu similar to the procedure presented earlier in the Changing Displayed Field Size section.

Move Field

You can move fields using the **Move Field** selection from the **Browse** pull-down menu similar to the procedure presented earlier in the Moving Fields section.

Resize Partitions

Activates the window splitter so that you can partition the window as described earlier.

Goto...

Displays a dialog so you can position the record pointer on a specific record in the browse window. ISAAC allows you to position the record pointer on a specific record in the browse window. FoxPro allows you to position the record pointer at the **Top** or **Bottom** record in the database, position on a certain **Record** number or **Skip** a certain number of records before positioning the record pointer.

Seek...

Displays the Expression Builder so you can search the active database to seek a particular field. Seek searches through the current field that the database was last sorted on (i.e., the index where your cursor is located). You must create an expression based on the index key. Character strings should be enclosed in single quotes, double quotes or square brackets. Numeric strings should be entered without any quotes. If ISAAC finds a matching value in the current index file, that database record will be selected.

For instance to search for NSN 1270123456789 in the Shopping List Data Window, you place your cursor on the NSN field and you select **Seek** from the **Browse** pull-down menu (see Figure 5-1). When the Expression Builder Window is displayed, then enter "127" to find the first NSN starting with 127 or enter "12701234576789." Figure 5-2 displays the Expression Builder window for this example. If there are multiple matching records ISAAC will select the first one that appears in the database order currently used. Seek is available on all windows with NSN related data (i.e., the **Component Data**, **Critical Item, Pipeline Data**, **Shopping List Data, View Input-Output** and **View Shop Comparison** Windows) plus the Multi-Day Evaluation Window.

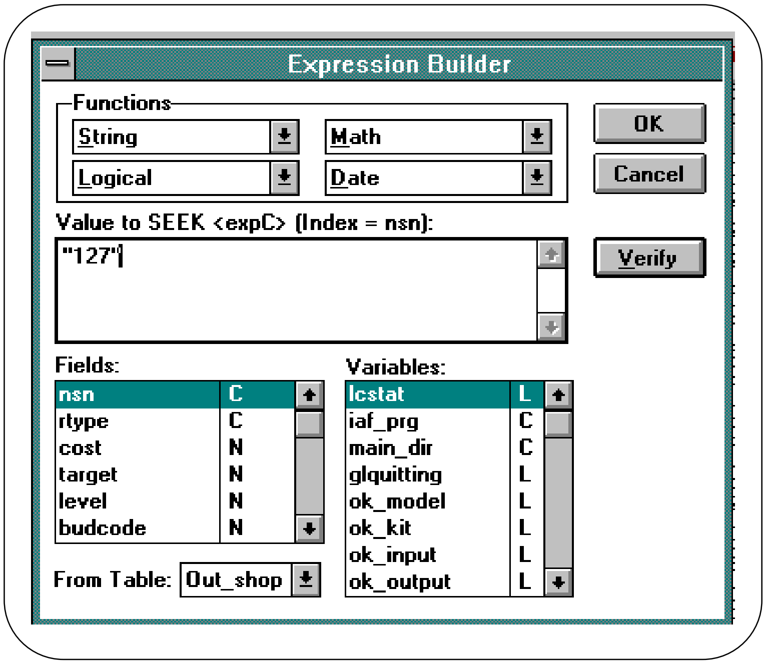


Figure 5-2
The Expression Builder Accessed Through the Seek Menu Selection

ISAAC creates a new index whenever the cursor moves from one field to the next. In our example, the cursor is on the NSN field, the database is sorted and indexed on the NSN field (the field indexed will be stated on the "Value to SEEK" line). You could also seek a particular cost value by first moving to the Cost field in the browse window and then seeking a specific value.

Toggle Delete

This capability is not present in either the modified or indexed browse windows.

Append Record

This capability is not present in either the modified or indexed browse windows.

PRINTING INPUT AND OUTPUT FIELD DATA

Input and output field data can only be printed via the respective picklist. When you select most of the view input, view output or miscellaneous pull-down menu options you may browse, print a report of, or preview a report of the respective data. (Statistics has no associated print option.)

We recommend that you preview a report prior to printing it. Each of the reports that has a **Print** option has the **Preview Report** option (except for the Performance Report). The model prints in a fixed font (it does not scale to fit the page) and will truncate all fields that do not fit on a page (from left to right). If you select **Preview Report** the model will display a window similar to Figure 5-3 (click the **Preview Report** box and then select the **Print** button):

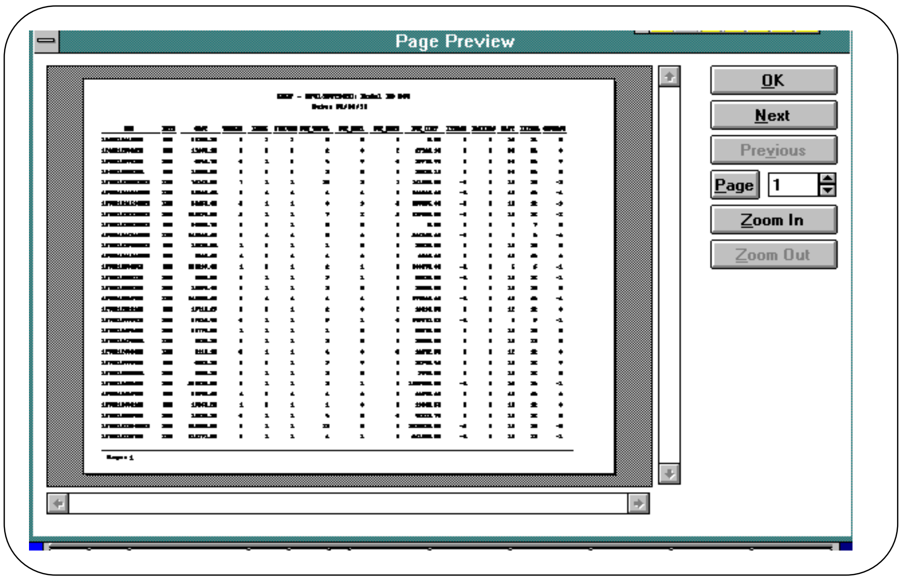


Figure 5-3
A Page Preview Sample

This window allows you to zoom in to view the text, to move forward or backward in the file one page at a time, and to move to a specific page. To close the window press **'Esc'** or click the control-menu box in the upper left corner of the window.

If you select the **Print** button a dialog box similar to Figure 5-4 will appear (with your default printer listed):

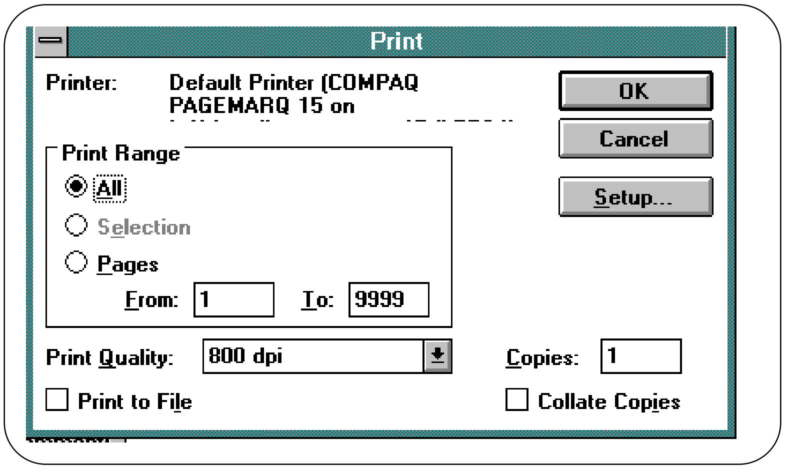


Figure 5-4
The Print Dialog Box

You may print the entire file (for the model run selected), select specific pages to be printed, or print to a text file. This print window is a standard Windows print dialog box. The model prints all reports in a fixed font (Courier) and uses the Landscape printing orientation.

You may print one or more sequential pages. If you are interested in printing data pertaining to different NSNs, we recommend that you use **Preview Report** to find the specific pages containing the data for those NSN’s. For many of the reports, you will probably need to use the **Zoom In** button to verify the NSNs on a given page. Many of databases that you may want to print from are fairly large (in terms of the number of pages of printed information). We recommend that you print several pages of a report on your printer, to get an idea of how long it takes to print the information, before you attempt to print an entire file.

Indexed Browse Windows

Several of ISAAC's windows have an additional capability that lets you sort the records using any field by clicking in that field. We term these windows indexed browse windows because it is the indexing of each field that permits the records to be sorted on multiple fields.

Applicable Windows

The following ISAAC windows are indexed browse windows. Each of these windows displays a large number of NSNs and therefore lends itself to being sorted in multiple ways.

1. **Component Data** Window from the **View Input** pull-down menu.

2. **Pipeline Data** Window from the **View Output** pull-down menu.

3. **Shopping List Data** Window from the **View Output** pull-down menu.

4. **Critical Item** Window from the **View Output** pull-down menu.

5. **View Input-Output** Window from the **Misc** pull-down menu.

Sequence Of Operation

The sequence of operation for each of the indexed browse windows is identical. It is presented in a generic form here as opposed to being presented in each of the respective window descriptions.

1. Select **one of the applicable windows from the respective** pull-down menu.

2. Select the fields you would like to see from the pick list presented. Choose the **Browse** button to view the fields selected. Browse and sort the data as appropriate. ISAAC will sort the records by any field that you click on or tab through. When you are finished, click the control menu box in the upper left corner of the window to close the window.

3. The pick list will be displayed again. You can again choose **Browse**, **Print** or **Cancel** (the window) as appropriate. If you choose **Print**, ISAAC will print the records in the sort order that existed when the window was closed.

4. You can cycle through steps 2 through 4 as many times as necessary. When you have completed what you want to do, select **Cancel** to return to the Model Parameters Screen.

General Characteristics

1. The default sort order for all of these windows is NSN order. Whenever you open a window for the first time the records will be sorted in NSN order.

2. The fields that are indexed for sorting are annotated with an asterisk to the left of the respective field check-box. The fields with gray asterisks are required fields in addition to being fields that are indexed for sorting. Those fields with magenta asterisks are indexed for sorting.

3. The sort order is always from lowest to highest numerical value with the lowest field value being displayed at the top of the window. Character fields are sorted similarly with the beginning of the alphabet at the top.

4. If you choose **Print**, ISAAC will print the records in the sort order that existed when the window was closed.

5. You can also seek any record with a particular value by placing your cursor in the field and choosing **Seek** from the **Browse** pull-down menu, and entering the value you wish to find as described earlier.

Temporary Databases

ISAAC creates a temporary database file when you browse any of the indexed browse windows so that you can export a single model run's data to other software (e.g., Microsoft Excel spreadsheet) for further analysis. The temporary files are written over if you view the same browse again.

ISAAC will create the following temporary database for the respective indexed browse window. Even though the Multi-Day Evaluation Window is not an indexed browse window, a temporary database is created when you browse it. Each of these databases is located in the iaf_port subdirectory of ISAAC:

1. Browsing the Component Data Window creates the out_kit.dbf temporary database.

2. Browsing the Critical Item Window creates the out_crit.dbf temporary database.

3. Browsing the Multi-Day Evaluation Window creates the out_mult.dbf temporary database.

4. Browsing the Pipeline Data Window creates the out_pipe.dbf temporary database.

5. Browsing the Shopping List Data Window creates the out_shop.dbf temporary database.

6. Browsing the View Input-Output Window creates the out_io.dbf temporary database.

Input Data Windows

Input consists of two separate windows: the **Component Data** Window and the **Run Log** Window. The **Component Data** Window displays item specific data for the current run while **Run Log** displays model run level input parameters for all runs. In both cases these windows allow the user to browse but not to modify the information contained on the respective window. The Input windows covered in this section each employ a ‘picklist’ where you select which fields you would like to browse.

Component Data

Component Data can only be selected through the **View Input** pull-down menu from the main menu. This selection allows you to browse the specific component data for both war and non-war conditions. It is the information developed by the **Kit** section of ISAAC. The item level input data pertains to the model run description displayed on the Model Parameters Screen. Clicking on the **Component Data** menu item will open the Choose ASMREC Fields picklist (Figure 5-5).

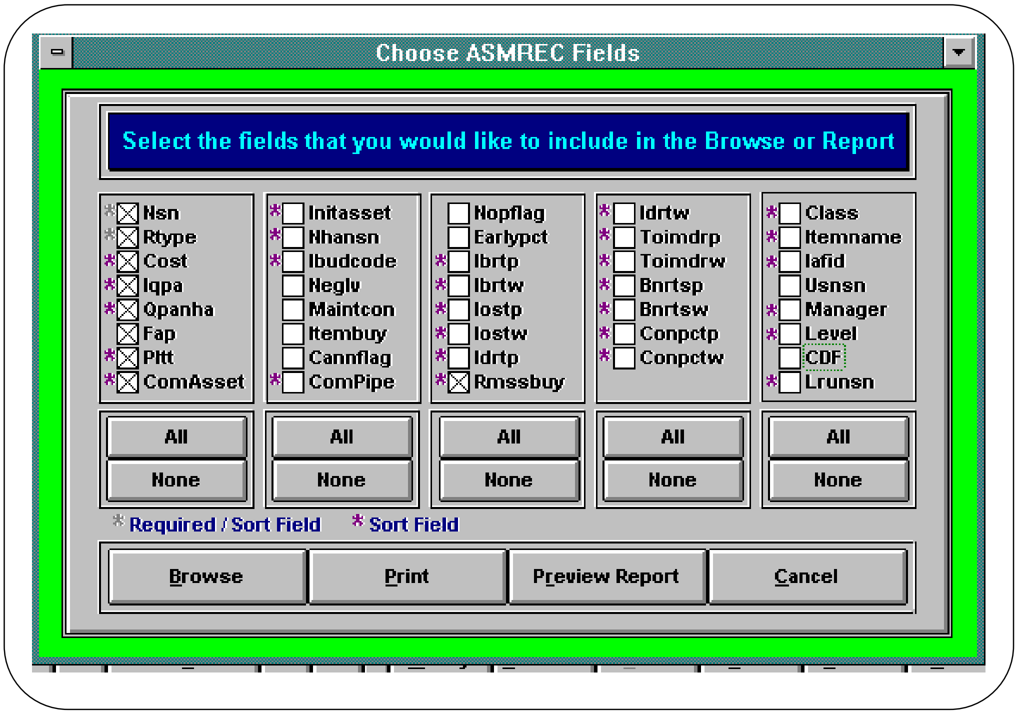


Figure 5-5
The Component Data Picklist

The fields displayed on this picklist are defined in Appendix B: Data Dictionary

The **Component Data** Window is an indexed browse window and therefore has a capability that lets you sort the records using most of the fields by clicking in that field. The fields that are indexed for sorting are annotated with an magenta asterisk to the left of the respective field check-box. The fields with gray asterisks are required fields in addition to be fields that are indexed for sorting.

If you select **Print**, ISAAC will then print the information in the sorted order. You can also seek any record with a particular value by placing your cursor in the field and choosing **Seek** from the **Browse** pull-down menu.

ISAAC creates a temporary database file when you browse **Component Data** so that you can export a single model run's data to other software for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_kit.dbf. The temporary files are destroyed if you view the same browse again.

Run Log

Run Log can only be selected through the **View Input** pull-down menu from the main menu. This selection allows you to browse all of the model run input parameters in the library of previous model runs. Clicking on the **Run Log** menu item will open the Choose PARAMS Fields picklist (Figure 5-6).

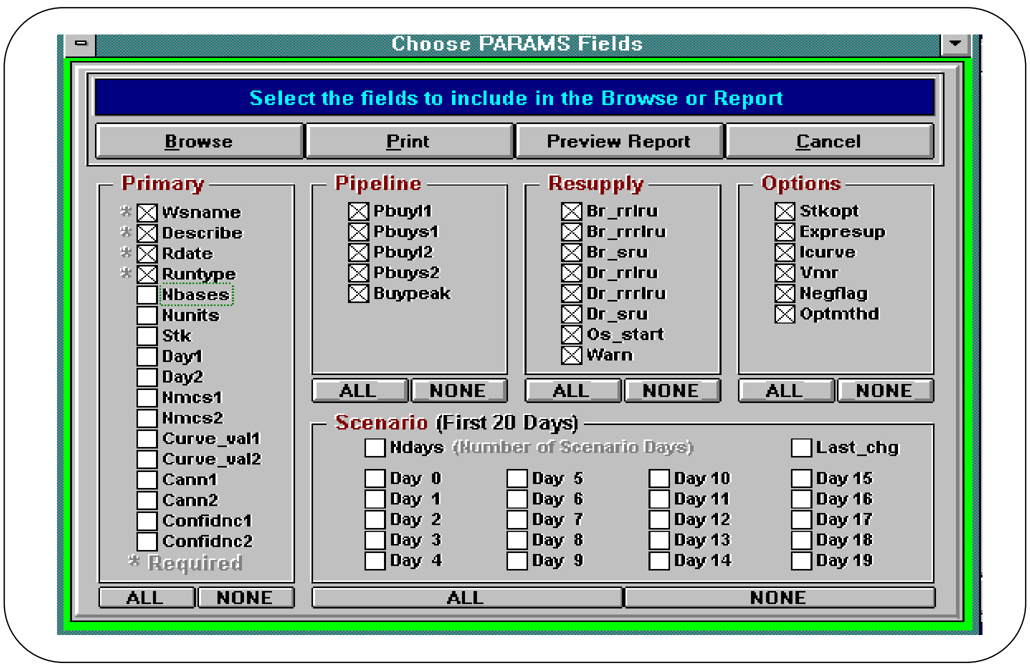


Figure 5-6
The Run Log Picklist

The fields displayed on this picklist are defined in Appendix B: Data Dictionary

Output Data Windows

Output is displayed in seven separate locations: the **Pipeline Data**, the **Shopping List Data**, the **Performance Report**, the **Curve**, the **Yearly Cost, Critical Item,** and **Multi-Day Evaluation** Windows. The **Performance Report** Window was covered in Chapter 2 and will not be addressed in this chapter. The **Critical Item** Window, **Pipeline Data** Window, and the **Shopping List Data**

Window display item specific data for the current run. The **Curve, Multi-Day Evaluation**, **Performance Report**, and **Yearly Cost** Windows display run level output for all items in the kit. The **Multi-Day Evaluation** Window will be covered in Chapter 8. In all cases these windows allow the user to browse but not to modify the information contained on the respective window. The output windows covered in this section (except for **Curve**) each employ a ‘picklist’ where you select which fields you would like to browse.

Pipeline Data

The **Pipeline Data** Window displays the pipeline mean (average broken items) at various locations (base, depot, order and ship) for each item and each analysis day (first and second analysis days on separate rows). It can be used to determine why the model recommended specific purchase quantities for each component.

Pipeline Data consists of component level pipelines, component cost, indenture information, and expected back order (EBO) data. It does not contain component order quantities or identify which components the model recommends for purchase.

Pipeline Data can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse the individual item pipeline data. This pipeline data will pertain to the model run description displayed on the Model Parameters Screen. Clicking on the **Pipeline Data** menu item will open the Choose ASMPIPE Fields picklist (Figure 5-7).

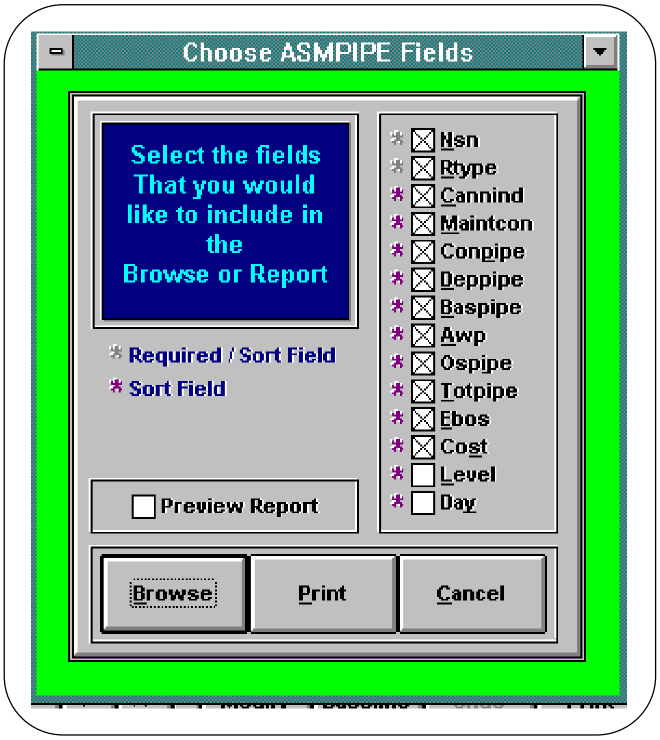


Figure 5-7
The Pipeline Data Picklist
The fields displayed on this picklist are defined in Appendix B: Data Dictionary

The **Pipeline Data** Window is an indexed browse window and therefore has a capability that lets you sort the records using any field by clicking in that field. If you select **Print**, ISAAC will then print the information in the sorted order. You can also seek any record with a particular value by placing your cursor in the field and choosing **Seek** from the **Browse** pull-down menu.

ISAAC creates a temporary database file when you browse **Pipeline Data** so that you can export a single model run's data to other software for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_pipe.dbf. The temporary files are destroyed if you view the same browse again.

Shopping List Data

Shopping List Data consists of component level buy cost, buy quantity information and recommended ordering timeframe as well as supporting data.

Shopping List Data can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse the individual item shopping list output parameters. This shopping list data will pertain to the model run description displayed on the Model Parameters Screen.

Clicking on the **Shopping List Data** menu item will open the Choose ASMSHOP Fields picklist (Figure 5-8).

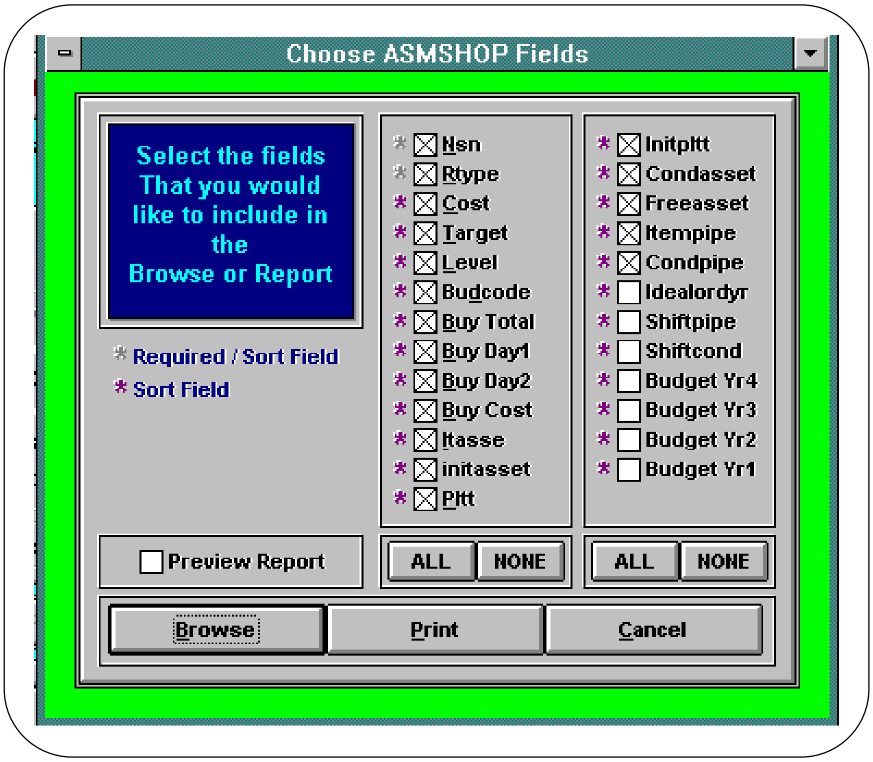


Figure 5-8
The Shopping List Data Picklist

The fields displayed in the Choose ASMSHOP Fields picklist are defined in Appendix B: Data Dictionary and are also described in the following section.

The **Shopping List Data** Window is an indexed browse window and therefore has a capability that lets you sort the records using any field by clicking in that field. If you select **Print**, ISAAC will then print the information in the sorted order. You can also **Seek** any record with a particular value by placing your cursor in the field and choosing seek from the **Browse** pull-down menu.

ISAAC creates a temporary database file when you browse Shopping List so that you can export a single model run's data to other software (e.g., Microsoft Excel spreadsheet) for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_shop.dbf. The temporary files are destroyed if ISAAC makes another run or if you view the same browse for another run.

The purpose of the **Shopping List** is to display the spares requirements solution and related item level information. The information presented in the Shopping List falls into 3 output categories. The first output category contains fields that specify the spares requirement target and order quantity. The second output category contains fields that specify which year the order is placed and ultimately the total spares buy cost for the weapon system. The final output category contains fields that specify how the model delays ordering some spares to more closely match annual budget constraints (see Model Approach section in Chapter 1). We will describe each category in turn.

The item fields relating to spares requirements are summarized by the following equation:

TARGET = BUY_TOTAL + ITASSE

- TARGET = total spares requirement needed to provide support through the end of the coverage period. The target equals the number of spares in inventory, maintenance, and transportation. These spares are required to replace the items that break in order to meet the aircraft availability target.

- BUY_TOTAL = the quantity of spares the model estimates for procurement (it is the sum of the two buy fields for each analysis day (BUY_DAY1 and BUY_DAY2).

- ITASSE = the total asset position at the end of the coverage period. The ITASSE is the sum of three related fields:

  - Free assets (FreeAsset) are the assets available from common components in already procured weapon systems in the IAF.

  - Initial assets (InitAsset) are other assets available from previous procurements.

  - Condemnations (CondAsset) are the model's accounting tool to estimate a part of the condemnations that occur over the coverage period. Condemnations are treated as negative assets that the model automatically buys. (The CondAsset value will be 0 for each item unless the model was run with the **Use Starting Assets?** drop-down list box set to **Use Assets: InitAsset +FreeAsset**. When CondAssets are not equal to 0 they are expressed as negative whole numbers.)

The item fields that specify the total procurement cost and the year the order is placed are the following:

- BUY_COST = the BUY_TOTAL multiplied by the item's unit cost (COST). Summing all BUY_COST will determine the model's estimated "Total Buy Cost" for the weapon system.

- IDEALORDYR = the year the item is ideally ordered. That year is the aircraft delivery year minus the PLTT (in years and rounded up). The model sums up all BUY_COST in a given IDEALORDYR to estimate annual spares buy costs. For instance, if an item has a PLTT of 14 months and the aircraft delivery year is 1998 then the IDEALORYR is 1996.

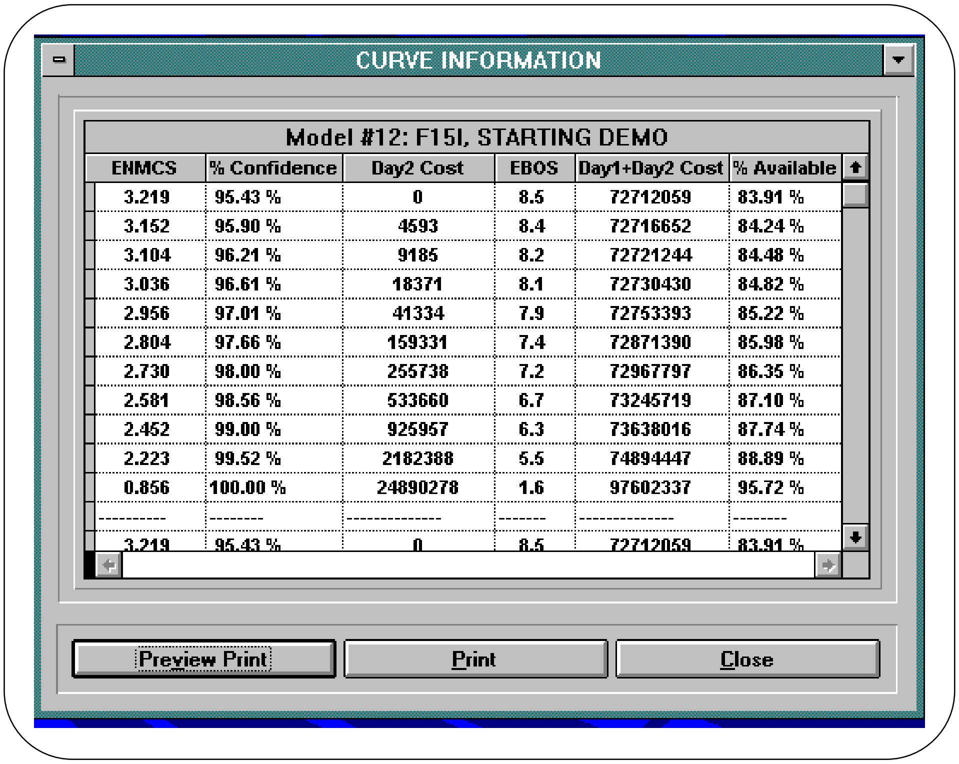
When the user enters an annual budget constraint on the Spares Cost Summary a PLTT before Spares Delivery Screen, the model delays some condemnations in order to match those constraints. The item fields that display the results of that matching process are the following:

– SHIFTCOND = the average number of condemnations that occur after the aircraft delivery year. [Note: the total condemnations in the coverage period equals (CONDPIPE + CONDASSET) so SHIFTCOND equals that total minus the first year's condemnations.] The SHIFTCOND spares are spares not required in the first coverage year so the model can delay ordering them without impacting the availability estimates. The model delays approximately the same percentage of SHIFTCOND across all items in an effort to match the annual budget constraint. (Note: the SHIFTCOND quantity is always less than or equal to the BUY_TOTAL.)

– Budget Years (e.g., 1995, 1996, 1997) = the spares order quantity for each year of the coverage period. If the user does not enter a budget constraint in the Spares Cost Summary, then the order quantity equals BUY_TOTAL and the budget year equals the IDEALORDYR. When the user enters a yearly budget constraint less than the annual buy costs (see Figure 1-9), the model will delay some or all of the SHIFTCOND spares so that the buy cost comes closer to the constraint. In that case, the item shows orders for as many as two years. For instance, an item may have some of the BUY_TOTAL ordered in 1996 (the IDEALORDYR) and some delayed and ordered in 1997 (some of the SHIFTCOND spares).

Curve

Curve can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse model run level cumulative cost and performance data. This curve data will pertain to the model run description displayed on the Model Parameters Screen. Clicking on the **Curve** menu item will open a Curve Window similar to Figure 5-9 (there is no picklist associated with this window).



Model #12: F15I, STARTING DEMO

ENMCS	% Confidence	Day2 Cost	EBOS	Day1+Day2 Cost	% Available
3.219	95.43 %	0	8.5	72712059	83.91 %
3.152	95.90 %	4593	8.4	72716652	84.24 %
3.104	96.21 %	9185	8.2	72721244	84.48 %
3.036	96.61 %	18371	8.1	72730430	84.82 %
2.956	97.01 %	41334	7.9	72753393	85.22 %
2.804	97.66 %	159331	7.4	72871390	85.98 %
2.730	98.00 %	255738	7.2	72967797	86.35 %
2.581	98.56 %	533660	6.7	73245719	87.10 %
2.452	99.00 %	925957	6.3	73638016	87.74 %
2.223	99.52 %	2182388	5.5	74894447	88.89 %
0.856	100.00 %	24890278	1.6	97602337	95.72 %
---------	--------	-------------	-------	-------------	--------
3.219	95.43 %	0	8.5	72712059	83.91 %

Figure 5-9
The Curve Window

The fields displayed above are described in the following section.

ENMCS — The total number of weapon systems (e.g., aircraft) that are expected to be not mission capable on Day 2 for a 2 day analysis or Day 1 for a 1day analysis, given the spares cost displayed in the Day 1 + Day 2 Cost column.

% Confidence — This is the probability of meeting the NMCS target, using the given Day 1 + Day 2 Cost, expressed as a percent.

Day 2 Cost — The incremental dollar cost of the components purchased to support the second analysis day for a 2 day run .

EBOs — The total expected back orders for all components for the analysis day.

Day 1 + Day 2 Cost — The sum of the Day 2 Cost value displayed plus the Day 1 Cost value (which is not shown; but, can be obtained by subtracting the Day 2 cost from the Day 1 + Day 2 Cost).

% Available — The percent of available aircraft equals the total number of aircraft under consideration minus the ENMCS aircraft with the result divided by the total number of aircraft under consideration.

To close the window press **'Esc'** or click the control-menu box in the upper left corner of the window.

Yearly Cost

Yearly Cost information consists of the aggregated component level buy cost and recommended ordering years as well as supporting data. (The individual component annual buy costs are broken out in the Shopping List Data Window.) Yearly cost data can be used to estimate future annual budget expenditures during each year of the coverage period. It can also be used, in conjunction with the specific component information from the Shopping List Data Window, to estimate the condemnation replacement purchase orders that can be delayed during each year of the coverage period.

Yearly Cost can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse the cumulative budget and annual cost output parameters. This yearly cost data will pertain to the model run description displayed on the Model Parameters Screen.

The **Yearly Cost** Window will display the results of your selections when the Spares Cost Summary a PLTT before Spares Delivery Screen was displayed at the conclusion of the particular model run that you have displayed on the parameters screen. Once you pressed the **F10** key when that screen was open, this database was populated with the data you can now view in this **Yearly Cost** Window.

Each of the fields in the **Yearly Cost** Window are described below. The field names (in parenthesis) are the names currently displayed on the **Yearly Cost** picklist and window. These fieldnames will be changed to the ones to the left of the parentheses so that this window will match the Spares Cost Summary a PLTT before Spares Delivery Screen.

Year (Idealordyr) — The year that components are placed on order.

Buy Cost (Cost Sum) — The cost sum of the components that need to be placed on order during the respective year so as to arrive in the aircraft delivery year. These components are being placed on order on the basis of their Procurement Lead Time Total (PLTT).

Budget — The user entered annual budget constraint.

Cost Adjusted (Cost_Adjus) — The model final Buy Cost after the model has adjusted the original buy quantity to try and match the budget.

Max Adjusted (Max Adjust) — Max. Adjusted represents the maximum number of dollars worth of spares that can have their orders delayed.

Delta Adjusted (Delta Adjust) — The dollar amount of the spares delayed for a specific year.

Critical Item List

The critical item feature of ISAAC generates a ranked list with items projected to ground the most aircraft at the top of the list. Since only the lack of a spare LRU directly grounds the plane, we first examine which LRUs are the most critical. Once these are identified, the user can then view the item's pipelines and SRUs, to better understand why the item is critical. The **Critical Item** Window presents a choice of basic views of the items:

– *LRU view* first presents all the LRUs with the most critical items at the top of the list. This view answers the question: "which LRUs potentially could ground the most aircraft?"

– *Detailed LRU and SRU view* presents the details of the critical LRUs. Typically the largest pipeline is the biggest contributor to criticality. ISAAC displays all the pipelines — base repair, depot repair, order and ship, and condemnation — as well as the AWP pipeline. If this is large that means SRUs are a problem. Thus, after each LRU, the browse displays all the SRUs (on all levels of indenture) on that LRU as well as the SRU pipelines and ranking.

The **Critical Item** list helps identify the item and the aspect that may be creating the problem. The model develops the critical item ranking for each analysis day the user selects. The **Critical Item** list does not necessarily identify what actions the user should take in order to make the item become less critical. Some critical LRUs may require an expedited repair at the depot or the base, others may need to have more assets procured, others may need to have SRU repair time expedited, others may have incorrect data inputs to the model, others may require further analysis, and so on. So the **Critical Item** list is useful in identifying the few items most critical from a large set of items and in quickly displaying possible areas for further examination.

In the next section, we first describe how we rank items and how an analyst can use those rankings to understand why items are critical and then we describe the actual browse and how the user can perform those analyses.

Critical Item Ranking

The model uses a different ranking scheme based on whether the maintenance policy is cannibalization or not. The model ranks items by EBOs for the no cannibalization case and by EBOs divided by the QPA for the cannibalization case.

Critical Items Under Cannibalization

In a cannibalization scenario, the model produces a rough estimate of the number of aircraft an item grounds as the ranking value — EBOs divided by IQPA. For example, if an item has 10 EBOs and IQPA of 3 (3 items or applications on the aircraft), 3.1 aircraft will be grounded because all backorders are consolidated on as few aircraft as possible with cannibalization. (In the critical item browse, the ranking value is termed**Lru_value**.) The greater that ratio, the greater the number of aircraft grounded and the more critical the item is.

How does one use this measure? Suppose the NMCS target is 2, i.e., the Air Force can tolerate two aircraft being grounded during war. Thus, any item that grounds more than 2 aircraft is a critical item because it is stopping you from reaching the goal of only 2 aircraft grounded. In our previous example of 10 EBOs for an item with a IQPA of 3, maintenance will consolidate all the backorders via cannibalization to 4 planes (10/3 rounded up). Note that for the 4th aircraft, a single backorder grounds the aircraft while the first 3 aircraft are grounded by 3 backorders each. Thus, one way of identifying the most critical item is selecting all items with ranking values greater than the NMCS target.

Critical Items with No Cannibalization

In the no cannibalization case, the model simply ranks items by EBOs and does not divide by IQPA. Because you do not cannibalize, 3 backorders may ground 3 planes even if the IQPA is three. The difficulty of the no cannibalization case is that with the simple EBO measure, the user can not determine which items are grounding more planes than the NMCS target. For instance, if the evaluation estimates you can not meet your NMCS target of 2, it is usually not because certain items are grounding more than 2 aircraft (as in the cannibalization case). It is usually because even though no item's EBOs are greater than 2 the cumulative spares shortage among all items grounds more than 2 aircraft. Thus, the items with the greatest EBOs are still the most critical but it is difficult to determine how critical. For this case, you might always assume the top 20 items (or top 5 percent) on the list are the most critical and since the browse gives each item a rank (termed **Lru_rank**), you can easily identify those items.

CRITICAL ITEMS WITH PARTIAL CANNIBALIZATION

With partial cannibalization you have a mixture of cannibalization and no cannibalization LRUs (the model always assumes SRUs are cannibalized). The model use EBOs as the value to rank the LRUs that are cannibalized and EBOs divided by IQPA as the value to rank the LRUs that are not cannibalized. Mixing those two cases together to produce a single ranking is misleading because in general the items that are not cannibalized have less flexibility and are more critical than the items that are cannibalized; however, it is difficult to estimate how much more critical. So the model leaves that judgment up to the user and simply identifies which items are cannibalized (**Cannflag** = Y) or not cannibalized (**Cannflag** = N).

Common SRU components

SRUs that are common to multiple LRUs on this aircraft are displayed in the browse under each parent LRU. That is if the SRU is on 3 LRUs, in the detailed view, the user sees the SRU displayed three times but each time the associated pipelines and EBOs are the aggregated impact of the SRU on the aircraft. The user can identify the common SRUs because their quantity on aircraft (the **Iqpa** browse field) is greater than the quantity on the LRU (the **Qpchain** browse field).

Critical Item Browse

Critical Item list can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse the critical item data. This critical item data will pertain to the model run description displayed on the Model Parameters Screen. Clicking on the **Critical Item** menu item will open the Choose Critical Items Fields picklist (Figure 5-10)

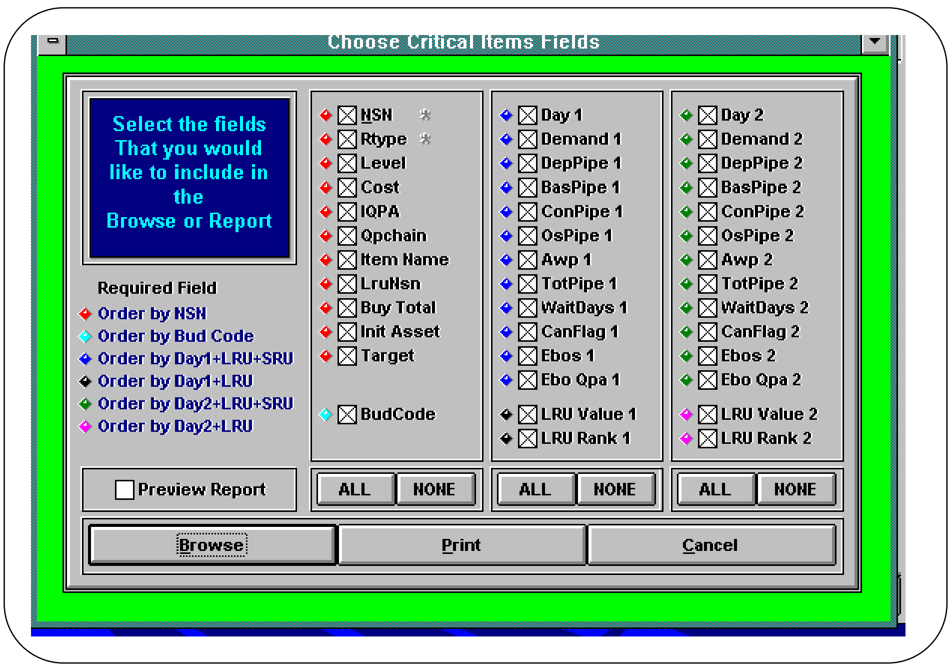


Figure 5-10
Choose Critical Items Fields picklist

The fields displayed in the Choose Critical Items Fields picklist are defined in Appendix B: Data Dictionary. Those fields that are unique to the Critical Items Window are also described in the following section.

The **Critical Item** Window is an indexed browse window and therefore has a capability that lets you sort the records using any field by clicking in that field. If you select **Print**, ISAAC will then print the information in the sorted order. You can seek any record with a particular value by placing your cursor in the field and choosing **Seek** from the **Browse** pull-down menu.

ISAAC creates a temporary database file when you browse **Critical Item** so that you can export a single model run's data to other software for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_crit.dbf. The temporary files are destroyed if you view the same browse again.

Field Groupings

The Choose Critical Items Fields picklist displays field names in 3 columns: 1) field names which end in "**1**" apply to the **1st Analysis Day** (column 2), 2) fieldnames which end in "**2**" apply to the **2nd Analysis Day** (column 3), and all other fields apply to both days (column 1).

The majority of the fields displayed in this browse window are identical to fields in other browse windows. The information comes from the **Component Data**, **Shopping List Data**, and **Pipeline Data** windows. The fields from **Component Data** maintain the same field names. The fields from **Shopping List Data** and **Pipeline Data** are similarly named; except in the **Critical Item** Window these fields have a _1 or _2 suffix, pertaining to the day of analysis. All of the fields contained in the **Critical Item** Window are defined in Appendix B. There are several fields which are unique to the **Critical Item** list. These fields are listed and defined as follows:

EBO_QPA_1 -- This field equals the EBOs field divided by the IQPA field for all cannibalization (CannFlag = 'Y') items. For non cannibalization items it is equal to zero.

EBO_QPA_2 -- Similar to EBO_QPA_1 but applies to Day 2.

LRU_VALUE1 — This is a nominal value assigned to each LRU based on that LRU's backorders. The higher the value the more critical the LRU is. This value is designed to provide a relative measure of criticality. For the cannibalization case, LRU_VALUE1 = EBOs / QPA (field labeled EBO_QPA_1). For the no cannibalization case, LRU_VALUE1 = EBOS_1 (field labeled EBO_QPA_1 equals 0). SRUs will have the same LRU_VALUE1 as their LRU.

LRU_VALUE2 — Similar to LRU_VALUE1 but applies to Day 2.

LRU_RANK1 — This field is used to rank the LRUs in order of their criticality. The lower the value the more critical the LRU is. This field is derived from LRU_VALUE1. The LRU with the highest LRU_VALUE1 will be assigned the LRU_RANK1 of "1." This is the most critical LRU. The next highest will be assigned a rank of '2" and so on. The ranking will be a continuous sequence and is only designed to provide an order of criticality not a measure of criticality. SRUs will have the same LRU_RANK1 as their LRU.

LRU_RANK2 — Similar to LRU_RANK1 but applies to Day 2.

Field Views and Indexes

The Choose Critical Items Fields picklist contains different colors in front of each field to identify the 6 different indexes available. Each index presents a different view of the critical items and has been designed for a different analytical purpose. The 6 different indexes belong to four basic groups of indexes. The first group consists of two indexes that present the LRU view — one each for the first and second analysis days. The model automatically adjusts the sort value of the index based upon whether cannibalization was the policy on that day or not. The second group consists of two indexes that each retain the same LRU ranking (as in the respective LRU only view); but, also adds the SRUs contained on the LRUs for each analysis day. The third group consists of one index that sorts the war day first by Budcode and then by LRU rank. The last group consists of one index that sorts all records on the basis of their NSNs. A more detailed description of the four groups of indexes or sort orders follows:

1. The LRU view *orders the LRUs* with the most critical items at the top of the list. For that view, the user selects the LRU_rank or LRU_value fields and clicks on those fields in the browse. If the user clicks on the rank or value for day 1 (e.g., Lru_rank_1) or day 2 (e.g., Lru_rank_2), the data is sorted on the basis of which items are most critical on the chosen day. The LRU with the highest Lru_value has the lowest rank and is displayed at the top of the list.

Critical Item - Model #65: F15ISUBSET: BASELINE DAYS 0 & 24

Nsn	Rtype	Level	Iqpa	Ebos_2	Ebo_qpa_2	Lru_value2	Lru_rank2
5865012112335EW	LRU	1	1	2.35	2.35	2.35	1
5865011799699EW	LRU	1	1	1.02	1.02	1.02	2
5865012876182EW	LRU	1	1	0.82	0.82	0.82	3
5895012973689FX	LRU	1	1	0.76	0.76	0.76	4
5841013468055FX	LRU	1	1	0.70	0.70	0.70	5
5841012905864FX	LRU	1	1	0.53	0.53	0.53	6
1270012684614	LRU	1	1	0.52	0.52	0.52	7
5865012905835EW	LRU	1	1	0.47	0.47	0.47	8
5985012778913FX	LRU	1	1	0.45	0.45	0.45	9
1270013133401FX	LRU	1	1	0.43	0.43	0.43	10
5841012843965FX	LRU	1	1	0.33	0.33	0.33	11
6680011066215	LRU	1	1	0.24	0.24	0.24	12
5895012965013EW	LRU	1	2	0.46	0.23	0.23	13
6620012320680	LRU	1	1	0.20	0.20	0.20	14
6605012400136FX	LRU	1	1	0.17	0.17	0.17	15
5895012731990	LRU	1	1	0.16	0.16	0.16	16

Figure 5-11
LRU Sort Order (with Cannibalization) for Analysis Day 2 — Sort Basis is EBOs

2. The detailed LRU and SRU sort order lists the LRUs and SRUs so that the LRU is displayed first and then followed by its respective SRUs. The LRUs are sorted in the same manner as in the LRU only view and contain their same rank. The SRUs contained on an LRU will be listed by level with the 2nd level SRU with the highest EBOs/IQPA at the top of the list of subordinate SRUs. All SRUs contain the same rank as their parent LRU. For that view, the user clicks any field with a 1 or 2 suffix except the Lru_rank or Lru_value fields.

Critical Item - Model #65: F15ISUBSET: BASELINE DAYS 0 & 24

Nsn	Rtype	Level	Deppipe_2	Awp_2	Totpipe_2	Ebo_qpa_2	Lru_rank2
5865012112335EW	LRU	1	2.72	1.61	8.00	2.35	1
5865004671140EW	SRU	2	2.55	0.00	2.98	0.66	1
5865004671191EW	SRU	2	2.25	0.00	2.63	0.48	1
5865011260430EW	SRU	2	3.91	0.00	4.57	0.33	1
5865011341082EW	SRU	2	1.70	0.00	1.99	0.21	1
5865012113991EW	SRU	2	2.54	0.00	2.97	0.16	1
5865012119086EW	SRU	2	2.36	0.00	2.76	0.14	1
5998011339953EW	SRU	2	0.45	0.00	0.58	0.14	1
5865011260429EW	SRU	2	0.40	0.00	0.47	0.09	1
6130004360579EW	SRU	2	0.35	0.00	0.45	0.09	1
5865011259547EW	SRU	2	0.79	0.00	0.92	0.08	1
5865011259548EW	SRU	2	1.09	0.00	1.28	0.05	1
5865011259546EW	SRU	2	0.43	0.00	0.50	0.02	1
5998011344682EW	SRU	2	0.20	0.00	0.26	0.02	1
5865011339951EW	SRU	2	0.31	0.00	0.36	0.01	1
5998004438714EW	SRU	2	0.13	0.00	0.17	0.01	1

Figure 5-12
LRU with Subordinate SRUs Sort Order for Analysis Day 2 (with Cannibalization) — Sort Basis is EBOs/QPA

3. The Budcode view orders all items in their budcode order. The secondary sort order is Lru_rank and the third sort order is Ebo_qpa of the SRUs contained on the LRUs. The budcode uses the first analysis day for the LRU rank unless the analysis day equals zero and then it uses the 2nd analysis day.

4. The NSN view orders all items by the NSN number. This is the initial sort order for the Critical Item Window. For that view, the user clicks the NSN field.

There are many ways to view the information. Figure 5-11 shows an LRU only view while Figure 5-12 shows an LRU with its SRUs view — both views for the second analysis day, under cannibalization. You can also show similar views for the first analysis day (whether under cannibalization or not). Either one of those views will allow you to see any of the fields on the **Critical Item** Window. In order to see the ranks of each item under both analysis days (e.g., the cannibalization and the no cannibalization policies) on one view, we recommend that you select a few of the fields on the picklist and leave the rest deselected. This is simpler than moving fields in and out of view to get the screen presentation that you want. Figure 5-13 is an example of a view presenting no cannibalization and cannibalization policies for analysis day 1 and 2, respectively.

Critical Item - Model #65: F15ISUBSET: BASELINE DAYS

Nsn	Rtype	Day_1	Lru_rank1	Day_2	Lru_rank2
5865012112335EW	LRU	0	38	24	1
5865011799699EW	LRU	0	8	24	2
5865012876182EW	LRU	0	10	24	3
5895012973689FX	LRU	0	1	24	4
5841013468055FX	LRU	0	2	24	5
5841012905864FX	LRU	0	4	24	6
1270012684614	LRU	0	3	24	7
5865012905835EW	LRU	0	12	24	8
5985012778913FX	LRU	0	5	24	9
1270013133401FX	LRU	0	6	24	10
5841012843965FX	LRU	0	7	24	11
6680011066215	LRU	0	91	24	12
5895012965013EW	LRU	0	9	24	13
6620012320680	LRU	0	89	24	14
6605012400136FX	LRU	0	19	24	15
5895012731990	LRU	0	23	24	16

Figure 5-13
Comparing No Cannibalization & Cannibalization Ranks

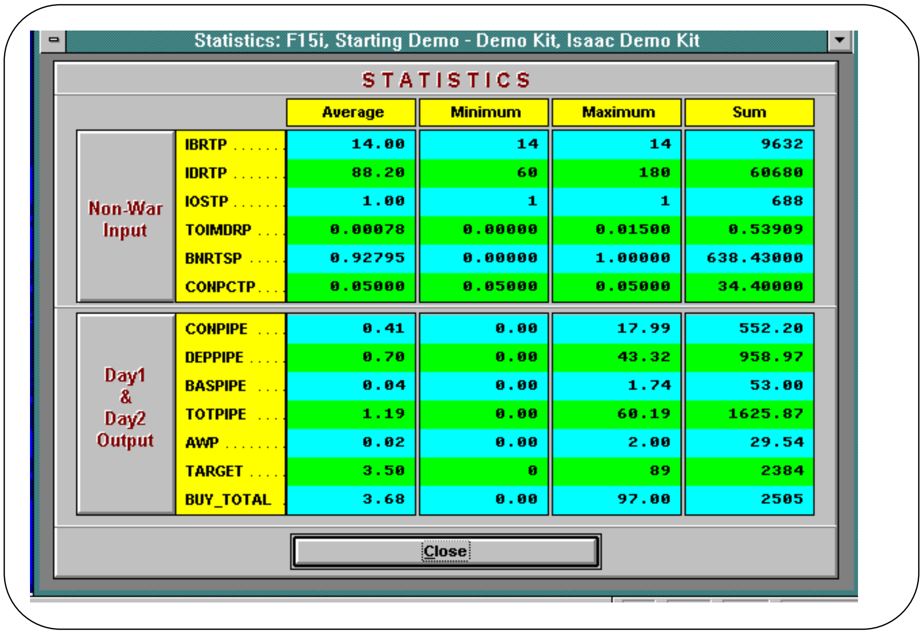
Notice, in Figure 5-13, the LRUs have different rankings on both days. In some cases the difference is very large (e.g., the top ranked LRU under the cannibalization policy is ranked number 38 under the no cannibalization policy). However, the 10 LRUs with the highest criticality under no cannibalization are among the top 13 seeded items of the criticality ranking under cannibalization.

Miscellaneous Data Windows

The **Misc** pull-down menu contains three choices. The first two choices, **Stats** and **View Input-Output**, contain a mixture of selected input and output summary parameters. **Stats** provides a summary of aggregated component level input and output statistics. **View Input-Output** provides specific component level input and output statistics broken out by NSN. The third choice, **View Shop Comparison**, allows you to compare selected individual item shopping list output parameters for two different model runs.

Statistics

Stats can only be selected through the **Misc** pull-down menu from the main menu. This selection allows you to view selected aggregated individual item input and output parameters. This statistical data will pertain to the model run description displayed on the Model Parameters Screen. Clicking on the **Stats** menu item will open a Statistics Window similar to Figure 5-14 (there is no pick-list associated with this window):



		Average	Minimum	Maximum	Sum
Non-War Input	IBRTP	14.00	14	14	9632
	IDRTP	88.20	60	180	60680
	IOSTP	1.00	1	1	688
	TOIMDRP	0.00078	0.00000	0.01500	0.53909
	BNRTSP	0.92795	0.00000	1.00000	638.43000
	CONPCTP	0.05000	0.05000	0.05000	34.40000
Day1 & Day2 Output	CONPIPE	0.41	0.00	17.99	552.20
	DEPPIPE	0.70	0.00	43.32	958.97
	BASPIPE	0.04	0.00	1.74	53.00
	TOTPIPE	1.19	0.00	60.19	1625.87
	AWP	0.02	0.00	2.00	29.54
	TARGET	3.50	0	89	2384
	BUY_TOTAL	3.68	0.00	97.00	2505

Figure 5-14
The Statistics Window

The table contains the minimum value, maximum value, average value, and the cumulative sum for each of the specified parameters. The parameters are broken out into two distinct groups. The first group (top portion of Figure 5-14) statistics pertaining to the non-wartime field values for the key input parameters that may vary on the basis of wartime and nonwartime conditions.

The Day 1 and Day 2 Output statistics presents the combined pipeline, target, and buy total for both analysis days (bottom portion of Figure 5-10). For runs with 2 analysis days, the pipeline statistics include information from both days while target and buy total include only one data point per item. The fields displayed are defined in Appendix B: Data Dictionary.

View Input-Output

Input-output data consists of component level indenture information, buy cost, quantity information, IAF managerial information, and summary pipeline data.

– It can be used to provide insight into why the model recommended specific purchase quantities for each component. Input-output data contains most of the salient information from the Shopping List Data Window and the total pipeline from the Pipeline Data as well as pertinent input information such as cost, indenture and quantity on the aircraft. Although it does not contain expected condemnations data, condemnations are the difference between the total buy quantity and the target quantity.

– Input-output data contains IAF managerial information (Budcode, Manager and IAFid) and can therefore be used to breakout component information in a number of ways to support item managers.

View Input-Output can only be selected through the **Misc** pull-down menu from the main menu. This selection allows you to browse selected individual item input and output parameters. This input-output data will pertain to the model run description displayed on the Model Parameters Screen (if viewed from the Model Parameters Screen). Clicking on the **View Input-Output** menu item will open the Choose Summary Fields picklist (Figure 5-15).

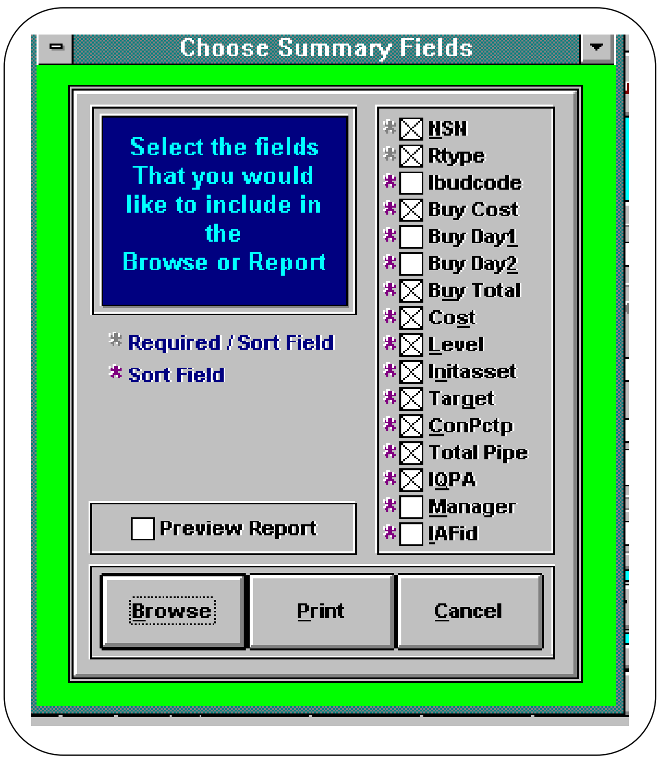


Figure 5-15
The View Input-Output Picklist

The fields displayed on this picklist are defined in Appendix B: Data Dictionary. The **View Input-Output** Window is an indexed browse window and therefore has a capability that lets you sort the records using any field by clicking in that field. If you select **Print**, ISAAC will print the information in the sorted order. You can also seek any record with a particular value by placing your cursor in the field and choosing **Seek** from the **Browse** pull-down menu.

ISAAC creates a temporary database file when you browse **View Input-Output** so that you can export a single model run's data to other software (e.g., Microsoft Excel spreadsheet) for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_io.dbf. The temporary files are destroyed if ISAAC makes another run or if you view the same browse for another run.

View Shop Comparison

Shop comparison data consists of component level buy cost, buy quantity and target quantity information for any two model runs in addition to the respective differences (deltas) between them in terms of cost, buy and target quantities. You can use this report to determine the differences by item of two model runs with different parameters, scenarios, or item data.

View Shop Comparison can only be selected through the **Misc** pull-down menu from the main menu. This selection allows you to compare selected individual item shopping list output parameters for two different model runs. This shopping list comparison data will pertain to the two model runs that you select from the View Shop comparison picklist. Clicking on the **View Shop Comparison** menu item will open the Shop List - Choose Two Model Runs for Comparison picklist window of previous runs similar to Figure 5-16.

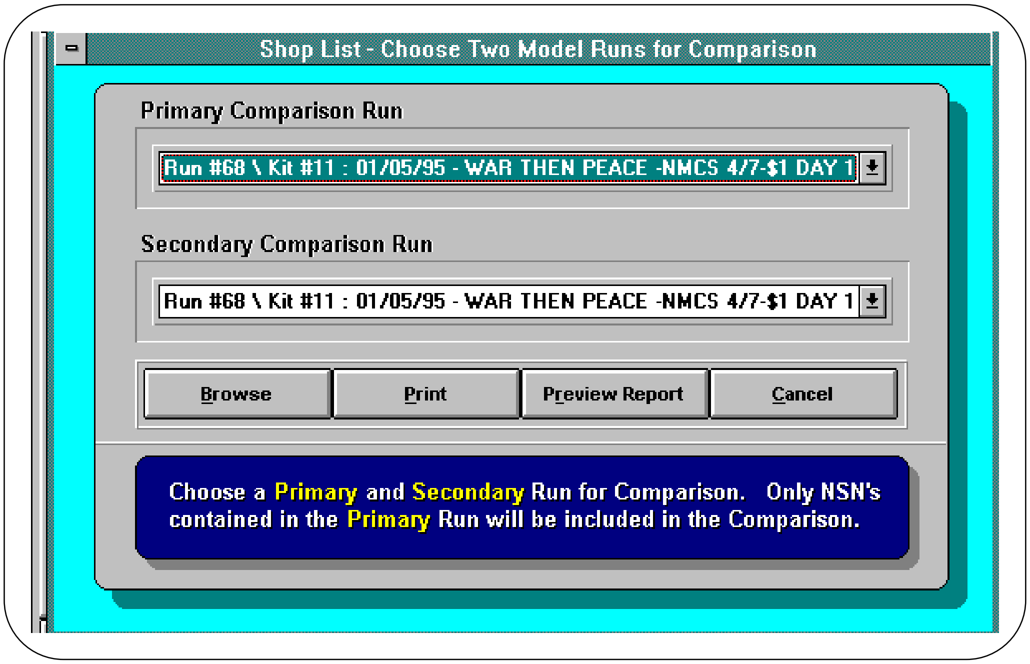


Figure 5-16
The View Shop Comparison Picklist

This type of picklist allows you to select which model runs you would like to compare. When you open this picklist, the default primary and secondary runs for comparison will be displayed in the Primary Comparison Run and the Secondary Comparison Run drop-down list boxes. The default for the primary and secondary comparison runs is the latest model run. To change the secondary comparison run (or the primary) click on the respective comparison run drop-down list box and scroll up or down to select the model run(s) you would like to compare.

This picklist does not allow you to select or deselect particular fields to browse or print. The default primary sort field for this window is the **Cost Delta** field. The individual item records will be sorted on the basis of the absolute value of the cost differential between the two runs (from highest to lowest). All of the deltas are recorded as absolute values.

When you have the two model runs selected that you would like to compare, choose the **Browse**, **Print** or **Preview Report** button as appropriate. Whichever you choose, the two model runs will be displayed with the NSNs contained in the primary model run displayed in the leftmost column (see Figure 5-17). The fields that pertain to the primary comparison run will contain the number 1 in the fieldname displayed at the top of each column (e.g., Targ1, Buytot1, etc.). Similarly, the fields that pertain to the secondary comparison run will contain the number 2 in the fieldname. The only components displayed in this window are those present in the primary comparison run. Therefore it is important to pick two model runs with some or all components (NSN values) in common.

Compare: (1) Run #8 - MULTIPLE BASES and (2) Run #12 - STARTING DEMO

Nsn	Targ1	Targ2	TargDelta	BuyTot1	BuyTot2	TotDelta	BuyCost1	BuyCost2	Cost Delta
5895012973689FX	24	7	17	25	8	17	15,539,475	4,972,632	10,566,843
5841012993515FX	27	5	22	28	6	22	9,307,816	1,994,532	7,313,284
5841013150646FX	14	5	9	16	7	9	11,685,328	5,112,331	6,572,997
1270013133401FX	36	9	27	41	14	27	8,005,496	2,733,584	5,271,912
5865011799699EW	34	12	22	34	12	22	7,266,718	2,564,724	4,701,994
5985012778913FX	16	5	11	19	8	11	6,891,319	2,901,608	3,989,711
5865012876182EW	21	9	12	21	9	12	5,199,033	2,228,157	2,970,876
5865012112335EW	43	6	37	43	6	37	3,159,781	440,899	2,718,882
5841012905864FX	11	4	7	14	7	7	5,149,018	2,574,509	2,574,509
5841012843965FX	12	5	7	14	7	7	4,958,772	2,479,386	2,479,386
5865012905835EW	30	11	19	30	11	19	3,441,390	1,261,843	2,179,547
5841012261171FX	12	2	10	12	2	10	2,377,680	396,280	1,981,400
6605012400136FX	26	7	19	30	11	19	3,002,040	1,100,748	1,901,292
5841013468055FX	5	3	2	6	4	2	4,718,442	3,145,628	1,572,814
5895012967183EW	0	2	2	0	2	2	0	1,490,714	1,490,714
6615013241821	13	5	8	16	8	8	2,742,064	1,371,032	1,371,032
6130013016341EW	21	4	17	21	4	17	1,686,750	321,285	1,365,465
5841013288234FX	11	2	9	11	2	9	1,614,478	293,541	1,320,937
5895013022076FX	8	1	7	8	1	7	1,496,624	187,078	1,309,546
1270012684614	4	2	2	5	3	2	2,682,155	1,609,293	1,072,862

Figure 5-17
View Shop Comparison Sample

If you select the **<u>P</u>rint** button the data will be printed with the primary sort order of Cost Delta and the secondary sort on Targ1 (the 1st day target spares quantity).

If you select the **<u>B</u>rowse** button the data will initially be sorted with the primary sort order of Cost Delta and the secondary sort on Targ1 (the 1st day target spares quantity). You can change the sort order by simply moving the cursor to the field that you would like to sort by and click on any of the data.

The item deltas are portrayed using their absolute values. This is done for sorting purposes, so that those components with the largest deltas (positive or negative) are displayed at the top of the screen. By sorting on the target deltas you can determine which items impacted availability the most. By sorting on the buy total deltas you can determined which items procurement quantities changed the most. Finally, by sorting on the buy cost delta you can determine which items impacted the budget the most. (Shop comparison data does not contain the quantities of each component assets expected to be condemned during the coverage period nor does it contain on-hand assets. However, their combined effect is the difference between the component’s buy quantity and target quantity).

ISAAC creates a temporary database file when you browse **View Shop <u>C</u>omparison** so that you can export a single model run's data to other software (e.g., Microsoft Excel spreadsheet) for further analysis. To create those files you must first browse the data. The separate dbf file is in the IAF_PORT subdirectory and is called pick2run.dbf (Shop Compare List). The temporary files are destroyed if ISAAC makes another run or if you view the same browse for another run.

Chapter 6

Text Editor and IPSS Interface

INTRODUCTION

This is a miscellaneous chapter that describes remaining model features. The ISAAC **Text Editor** enables you to view text files that you can not access through the browse windows of ISAAC. The **IPSS** interface is the way to format ISAAC output for export to the Initial Provisioning Support System (IPSS). IPSS is the mechanism the IAF uses to order the parts that ISAAC determines are required.

THE ISAAC TEXT EDITOR

The ISAAC **Text Editor** can be accessed from the initial ISAAC screen, the Model Parameters Screen, or one of the browse windows accessed through the **View Input**, **View Output**, or **Misc** pull-down menus. The editor can not be accessed from the Kit Parameters Screen, when any of the picklists are in view, or when the IPSS window is open.

The **Text Editor** enables you to open existing files but not to create new files. For instance, you can review the import errors that the model identified with the text editor or create simple text file with user notes. The editor can be used to modify the appearance of the text files in terms of font size, spacing, indention, etc. However, in the text editor mode you can modify the text. **Text Editor** should not be used to modify program or executable files!

The **Text Editor** pull-down menu (Figure 6-1) contains eight different choices. The first six choices are specific file types. The seventh choice (Unlisted Files) is not restricted to a specific file type. The last choice (Preferences) is deselected until you have selected a file type and opened a specific file.

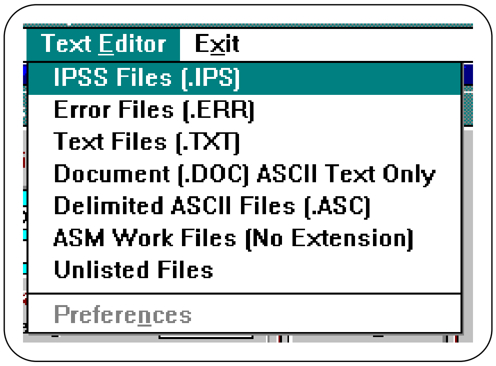


Figure 6-1
Text Editor Pull Down Menu

The menu has been set up in this way so that you can quickly view files of a certain type. For instance, if you are looking for error files the text editor will take you to the work subdirectory and list all the error files automatically by selecting **ERROR Files (.ERR)** from the pull-down menu. Once you select a file you will need to expand the size of the window and turnoff the word wrap function (see preferences next) to view a large tabular file.

The following is a list of the **Text Editor** pull-down menu choices with the type of file they are formatted to print to the screen as well as the respective subdirectory they access directly.

1. **IPSS Files (.IPS)** — are files formatted for export to the Initial Provisioning Support System (IPSS). This choice takes you to the iaf_port subdirectory.

2. **Error Files (.ERR)** — are files that list specific model error messages. This choice takes you to the work subdirectory.

3. **Text Files (.TXT)** — are files used for ASCII loading. This includes input files that are formatted for import into ISAAC. This choice takes you to the iaf_port subdirectory.

4. **Document (.DOC) ASCII Text Only** — can be notes to yourself and read-me files from LMI. This choice takes you to the work subdirectory.

5. **Delimited ASCII Files (.ASC)** — are files that are similar to the .sdf files but that are delimited by commas. This choice takes you to the primary ISAAC directory.

6. **Work Files (No Extension)** — are files created by the ASM within ISAAC. This choice takes you to the work subdirectory.

7. **Unlisted Files** — are all other files. This choice takes you to the primary ISAAC directory.

8. **Preferences** — The preferences option provides the user with several edit preference options. The most useful option is the **Wrap Words** preference. By deselecting **Wrap Words** you can view records that are wider than 80 columns without the records wrapping around. The preferences option will be deselected until you select a file type and open a file.

Opening and Closing Files

This section will describe how to open files using the text editor, how and why to use the Edit Preferences, and how to close files using the text editor.

Choosing any one of the pull-down menu choices will display an Open dialog box similar to Figure 6-2. This is a standard Windows Open dialog box.

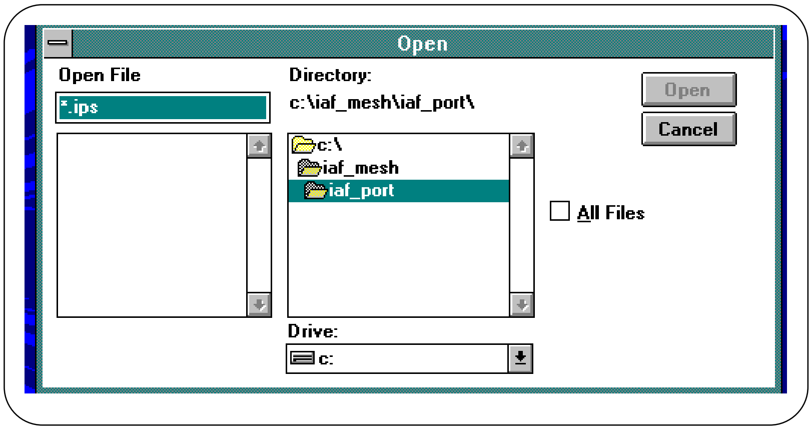


Figure 6-2
The Open Dialog Box

Opening Files Using the ISAAC Open Dialog Box

To open a file:

1. From the main menu select **Text Editor** and select the file type you are interested in.

2. The Open dialog box appears.

3. If the file you want to open is on a different drive, select the drive you want from the **Drive:** drop-down list box.

4. If the file you want to open is on the same drive but in a different directory or subdirectory, select the directory and or subdirectory you want from the **Directory:** list box. (Double-click the directory, or press the UP ARROW or DOWN ARROW key to select the directory, and then press ENTER).

5. ISAAC displays the names of all files in that directory that are the type selected in the Open file text box. To display a different type of file, enter the appropriate file extension in the Open file text box or you may click on the **All Files** check box to display all files in the directory.

6. From the list of files, select the file you want to open.

7. Double-click the filename or choose the **Open** button.

Edit Preferences

The Edit Preference window (Figure 6-3) enables you to modify save options, justification and several edit and display options. The **Preferences** pull-down selection will be deselected until you have opened a file using the **Text Editor** pull-down menu.

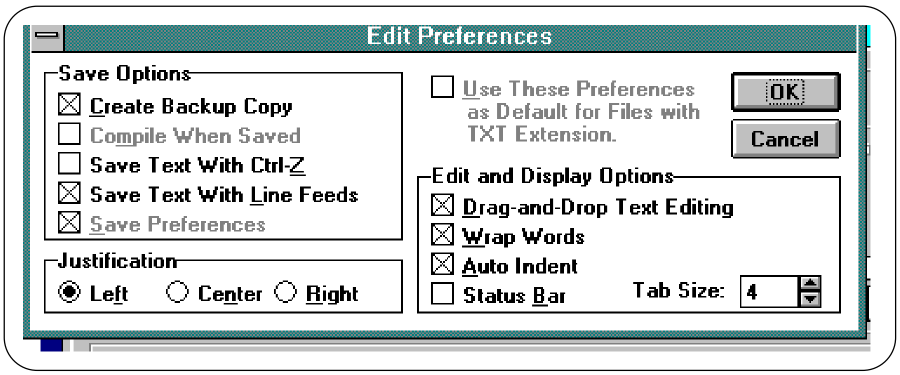


Figure 6-3
The Edit Preference Window

The preferences that are checked when the Edit Preferences dialog box is opened are the default selections. We recommend that you deselect the **Wrap Words** option. When **Wrap Words** is selected, the editor displays the entire record in the view window. When it is deselected only a part of each record is displayed in the view window. However what is displayed will be in a tabular format. For many ISAAC files, that are greater than 80 columns wide, the file will wrap around when it is first opened. If this is the case deselect **Wrap Words** and you can view file without the records wrapping around.

Closing Files

To close an open text file that you have not changed press the **Esc**' key or close the file using the control menu box in the upper left corner of the window.

To close an open text file without saving any changes (after you have made changes), close the file using the control menu box in the upper left corner of the window and select the **No** button in the Do you want to save changes in the [filename]? dialog box.

If you want to save your changes, close the file using the control menu box in the upper left corner of the window and select the **Yes** button in the Do you want to save changes in the [filename]? dialog box.

Text Options

The **Text** option only appears on the main menu.

Selecting **Text** from the main menu opens the **Text** pull down menu (Figure 6-4).

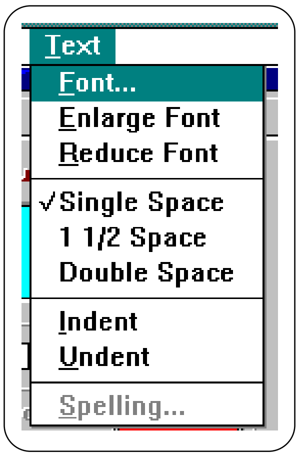


Figure 6-4
Text Pull Down Menu

Each of the eight active options are described below. The last option (Spelling) is not currently active.

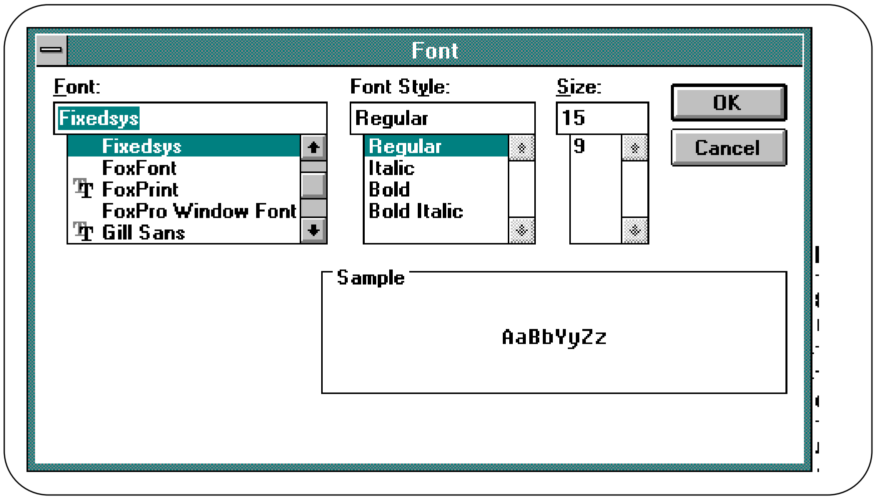


Figure 6-5
Font Dialog Box

1. **Font...** — Choosing **Font...** from the pull-down menu opens the Font dialog box (Figure 6-5). You can vary the font, font style and font size using this dialog box.

2. **Enlarge Font** — This selection will enlarge the font uniformly for all text on the entire page (not just highlighted text). It is a toggle switch with **Reduce Font**.

3. **Reduce Font** — This selection will reduce the font uniformly for all text on the entire page (not just highlighted text). It is a toggle switch with **Enlarge Font**.

4. **Single Space** — This is the default spacing for the entire document.

5. **1 1/2 Space** — This will modify spacing to 1 1/2 space for the entire document.

6. **Double Space** — This will modify spacing to double spacing for the entire document.

7. **Indent** — This will indent highlighted text or text to the right of the cursor approximately 1/2 inch. Every time you choose **Indent** the selected text will move 1/2 inch to the right.

8. **Undent** — This will "undent" highlighted text or text to the left of the cursor approximately 1/2 inch. Every time you choose **Undent** the selected text will move 1/2 inch to the left.

9. **Spelling** — Not currently active.

ISAAC TO IPSS INTERFACE

The **IPSS** interface is the way to format ISAAC output for export to the Initial Provisioning Support System (IPSS).

Relationship Between ISAAC and the IPSS

The IPSS provides input to ISAAC accepts ISAAC output and is the mechanism for ordering the parts. IPSS collects item level information from MALHA and has access to US MIL STD information. IPSS formats the input data so that it can be imported into ISAAC. ISAAC is used to calculate the spares provisioning requirements for a particular weapon system. ISAAC output is formatted for export to the IPSS. The actual spares orders are placed though the IPSS.

Preparing ISAAC Output for Export to the IPSS

When you select **IPSS** from the main menu bar a window similar to Figure 6-6 will appear.

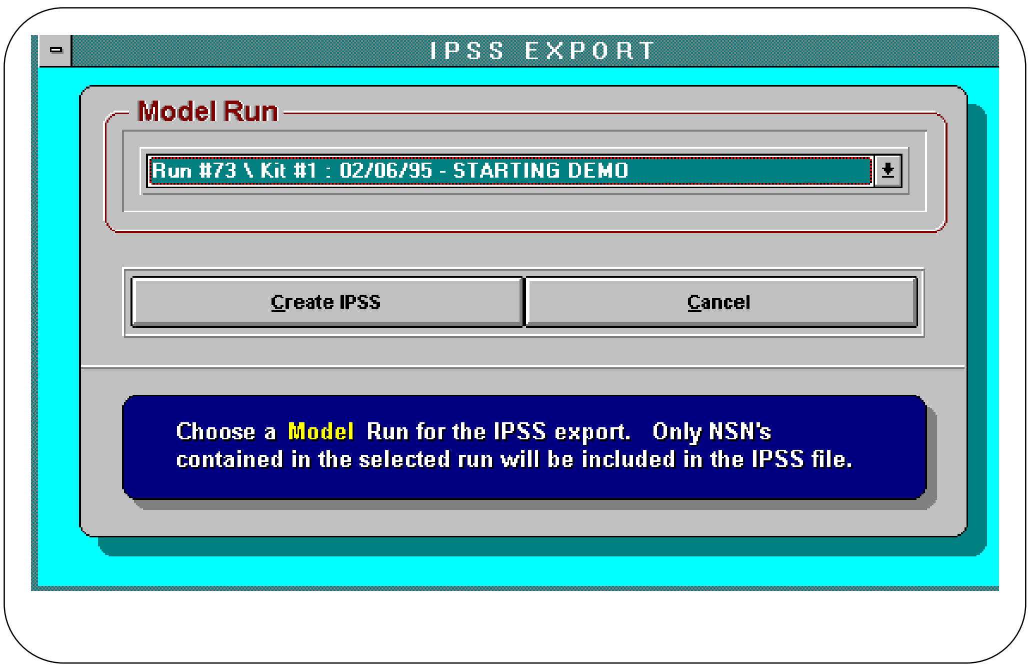


Figure 6-6
IPSS Export Window

The model run number displayed in the Model Run box of the IPSS Window is the most recent model run not necessarily the model run that is displayed on the Model Parameters Screen. If you want to select another model run to export to IPSS format, click on the Model Run box and you can either scroll up or down to the desired run and press **Enter**' or click on the desired model run with the left mouse button.

When you have the model run you want to export displayed in the Model run box, choose the **Create IPSS** button and a Save As dialog box similar to figure 6-7 will appear on the screen.

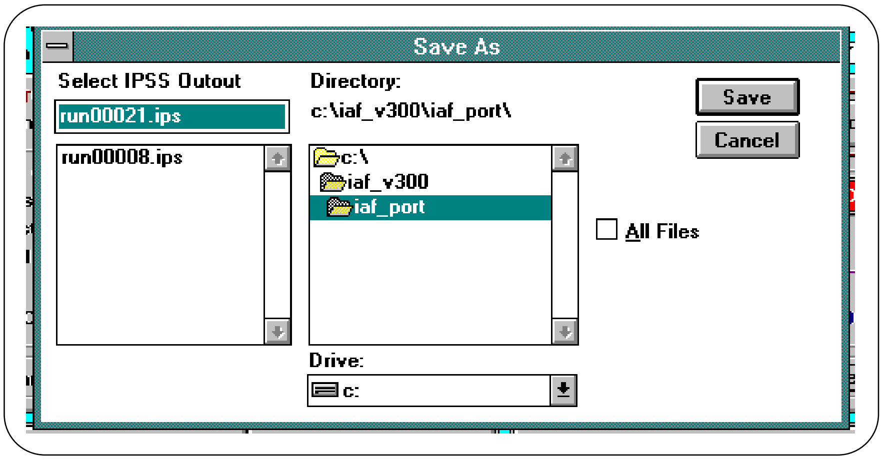
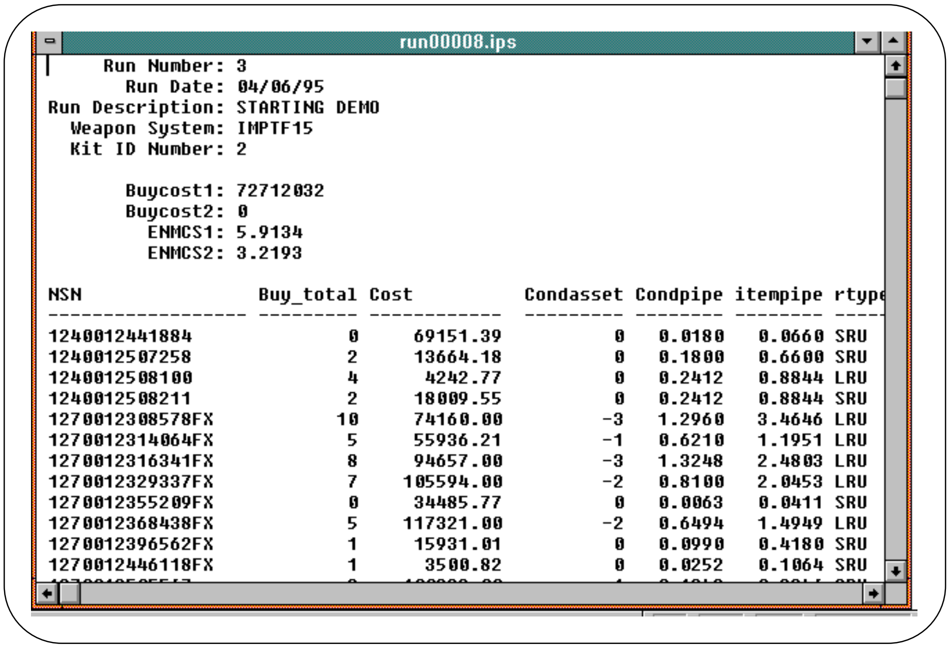


Figure 6-7
Select IPSS Output Window

ISAAC uses the model run number and automatically creates an output file name based on that run number. The default directory that ISAAC will place the file in is the iaf_port subdirectory. If the file name and subdirectory location are acceptable, choose the **Save** button on the Save As dialog box. This is all that is required to create the exportable file. Your output will be reformatted to IPSS format and a window similar to figure 6-8 will be displayed.

If you want to use a more descriptive filename and / or change the subdirectory, directory or drive location, you can enter the desired 8 character filename (with a .ips extension) and change the storage location using standard Windows Save As dialog box techniques. Once you are satisfied, choose the **Save** button.



run00008.ips

```
     Run Number: 3
       Run Date: 04/06/95
Run Description: STARTING DEMO
  Weapon System: IMPTF15
  Kit ID Number: 2

      Buycost1: 72712032
      Buycost2: 0
        ENMCS1: 5.9134
        ENMCS2: 3.2193
```

NSN	Buy_total	Cost	Condasset	Condpipe	itempipe	rtype
1240012441884	0	69151.39	0	0.0180	0.0660	SRU
1240012507258	2	13664.18	0	0.1800	0.6600	SRU
1240012508100	4	4242.77	0	0.2412	0.8844	LRU
1240012508211	2	18009.55	0	0.2412	0.8844	SRU
1270012308578FX	10	74160.00	-3	1.2960	3.4646	LRU
1270012314064FX	5	55936.21	-1	0.6210	1.1951	LRU
1270012316341FX	8	94657.00	-3	1.3248	2.4803	LRU
1270012329337FX	7	105594.00	-2	0.8100	2.0453	LRU
1270012355209FX	0	34485.77	0	0.0063	0.0411	SRU
1270012368438FX	5	117321.00	-2	0.6494	1.4949	LRU
1270012396562FX	1	15931.01	0	0.0990	0.4180	SRU
1270012446118FX	1	3500.82	0	0.0252	0.1064	SRU

Figure 6-8
Sample of IPSS Export File Produced from ISAAC Output

Data that is Exported to IPSS

The information that will be exported falls into two categories: model run summary data and item-level model output data.

Model Run Summary Data

Buycost 1 — This is the combined cost of the total number of each respective component purchased for the first analysis day.

Buycost 2 — This is the combined cost of the total number of each respective component purchased for the second analysis day.

ENMCS 1 — The number of weapon systems (e.g., aircraft) that are expected to be not mission capable, due to a shortage of spares, on the first analysis day.

ENMCS 2 — The number of weapon systems (e.g., aircraft) that are expected to be not mission capable, due to a shortage of spares, on the second analysis day.

Kit ID Number — The kit identification number for the input data that was used in the model run.

Run Number — The ISAAC model run number that the exported information comes from.

Run Description — The user entered run description (from the Model Parameters Screen).

Run Date — The date of the ISAAC model run.

Weapon System — The weapon system (from the next higher assembly number) for the LRUs.

ITEM LEVEL MODEL OUTPUT DATA

Budcode — A user defined budget code from 1 to 99 which is used to record composite output parameters for a group of items (such as subsystems, LRUs, SRUs, Consumables etc.,).

Buy total — This is the total spares quantity that the model recommends purchasing by component.

Buy_day1 — The total number of spares purchased for the first analysis day by component.

Buy_day2 — The total number of spares purchased for the second analysis day by component.

Condasset — The condemnations during the coverage period minus the items expected to be condemned during the items procurement lead time (PLTT). The condasset value will be 0 for each item unless the model was run with the **Use Starting Assets?** field toggled to **Use Assets: InitAsset + FreeAsset**. When Condassets are not equal to 0 they are expressed as negative whole numbers.

Condpipe — Condemnation pipeline is the number of components expected to be condemned during the item's procurement lead time (PLTT).

Cost — Unit cost of the component in US Dollars.

Free asset — Freeassets is an ISAAC specific term developed to incorporate common components into the spares calculation. Common components are components common to both the initial provisioning aircraft and other IAF aircraft. Treatment of common components must apply any surplus stock already in the IAF inventory toward the requirement and must consider economies of scale. Since stock is already available for other aircraft, the new aircraft needs less inventory than if the part were not common. ISAAC uses a simple approximation to incorporate those benefits that are expressed in terms of "free assets": assets available to the item free of charge because of its commonality characteristics.

Initasset — The starting asset position for the NSN before any buys are made by ISAAC.

Initpltt — Procurement leadtime for the item in months. The initpltt is the time from when an item is condemned to when a serviceable replacement for the item is procured and available at the base.

Itasse — In general, itasse is the initial asset position plus any free assets minus any condemnations that are expected to occur during the coverage period.

Itempipe — This is the sum of the base pipeline plus the depot pipeline plus the order and ship pipeline and the condemnation pipeline from the ASMPIPE database by component for NON-WARTIME ONLY.

Level — Indenture level of the component. The indenture structure describes the relationship between components.

NSN — The number used to identify each specific component.

Rtype — The type of reparable this spare is classified as (i.e., LRU or SRU).

Target — This is the number of spares purchased by component minus those spares that are expected to be condemned during the coverage period.

Chapter 7

Programs that must be Executed Outside of ISAAC

INTRODUCTION

This chapter describes the purpose and execution of the two programs that are associated with ISAAC: 1) the *ISAAC Filer* that is used to archive and retrieve ISAAC databases and 2) *Set Keys* that is used to reinitialize the ISAAC databases in the event of a power failure or windows interruption. Both are contained in the same Windows folder (program group) where ISAAC is accessed. Note: the ISAAC Filer and Set Keys must be operated with ISAAC closed. We also describe how to install these programs separately from ISAAC.

FILER FOR ISAAC

ISAAC filer allows the user to manipulate all the previous ISAAC runs and baseline kits, the ISAAC data bases. Specifically the filer allows the user to backup existing databases, create multiple locations to store databases, and retrieve other databases. Note, to reduce hard disk usage, Filer automatically condenses ("Zips") the databases using PKZIP on all archives operations and unzips the databases on all retrieves.

ISAAC Filer Options

The basic options for the filer are to archive and/or retrieve a database. Those two operations individually or in combination enable the user perform the following operations:

ARCHIVE

With the Archive option (left hand screen of Figure 7-2), the user can perform three basic operations:

1. Create an archive copy of the current ISAAC database.

2. Continue to backup to the same archive copy when ISAAC databases change.

3. Export the database to multiple "floppy" disks to transfer to another PC with ISAAC or for additional backups.

RETRIEVE

With the Retrieve option (right hand screen of Figure 7-2), the user can perform four basic operations:

1. Delete the existing data base by retrieving another database and overwriting the existing database.

2. Retrieve previous work stored in an alternate archived database.

3. Import the baseline database. This lets the user start over with only one previous run and baseline kit.

4. Import the database from multiple floppy disks created by the same or another PC running ISAAC.

Using the ISAAC Filer

ISAAC Filer is accessed by double clicking the **ISAAC Filer** icon (Figure 7-1) in the Windows folder where ISAAC is maintained.

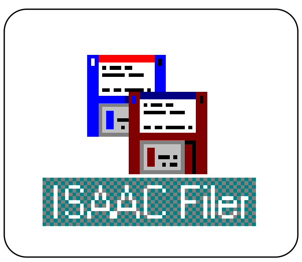


Figure 7-1
The ISAAC Filer Icon

Double clicking the ISAAC Filer Icon will open the ISAAC Filer Screen (Figure 7-2).

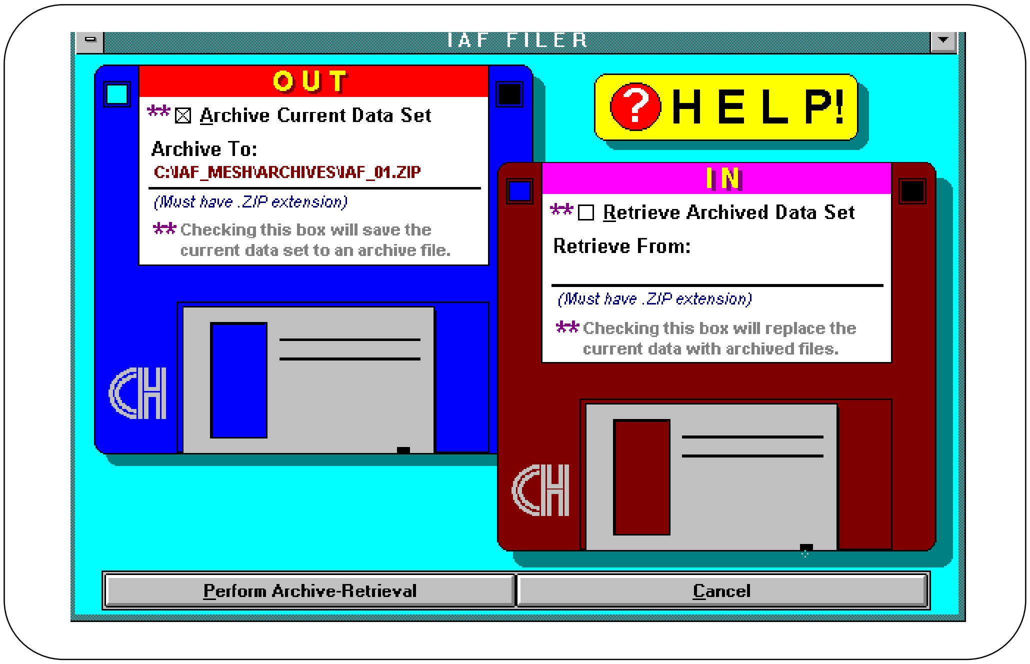


Figure 7-2
The IAF Filer Screen

The Archive or "Out" part of the screen displays the most recently archived file. To specify a different Archive file you perform the following steps:

1. Click on the line where the file and directory and a dialog box file picker opens (you can also click on the box titled *Archive to current data set* to get the file picker).

2. That file picker lets you select a drive, directory, and specify a file name in the standard Windows fashion (see Figure 6-1 for an example). Note: the default subdirectory is called "Archives".

3. Once you specify a file and location click on the **Save** button.

4. If you only want to archive your data, then click on the **Perform Archive-Retrieval** Button.

By checking the box for the Retrieve option (right hand screen), you then can select the location and the name for the archived file for retrieval (similar to the archive process just described). Once selected, the file name appears in the Retrieve Screen.

By entering the archive and retrieval information, the user can swap databases. For instance, to save the current runs for a particular analysis and start over for the next analysis, archive the current database in "Analysis1.zip" and retrieve the "baseline.zip".

Note, to reduce hard disk usage, Filer automatically condenses ("Zips") the databases using PKZIP on all archives operations and unzips the databases on all retrieves. Thus, the filer can only retrieve databases created via archive operation, because they contain a specific format.

SET KEYS

Set Keys is used to reinitialize the ISAAC databases in the event of a power failure, machine shut-down while ISAAC is in operation, or other windows events that cause improper closing of a windows application. Set keys reinitializes and re-indexes all databases that might be corrupted. You will know if a database is corrupted because a corrupted database will cause an immediate error when entering ISAAC.

Using Set Keys

Set Keys is accessed by double clicking the **Set Keys** icon (Figure 7-3) in the windows folder where ISAAC version 3.1 is maintained.

Figure 7-3
The Set Keys Icon

Set Keys in Operation

Double clicking the **Set Keys** Icon will execute the Set Keys program. Set keys will reinitialize and re-index your ISAAC databases. Note: ISAAC must be closed before you execute Set Keys. The user does not need to perform any other operations.

INSTALLATION

This section describes how to install the other ISAAC related program icons if they were not already installed when ISAAC was installed. The following steps are for both programs (with a few differences that we will point out):

- Start windows and open the program manager.
- Click on the program group that contains the ISAAC Icon.
- Select **File** from menu bar (top left hand corner).
- Select **New** from the **File** pull-down menu.
- Choose the **Program Item** radio button.
- The insertion point will be at the **Description:** text box. Type in "**ISAAC FILER**" or "**Set Keys**" (as appropriate).
- Click in **Command Line:** text box then:
  - Click on the **Browse** button.
  - Move to C:\ISAAC.
  - Click on the appropriate executable file - FILER.EXE or SETKEYS.EXE.
- Click in the **Working Directory:** text box and type - "**C:\ISAAC**".

Once you are done with the ICON installation the program manager program group should have the appropriate icon as displayed earlier in this chapter.

Chapter 8

Evaluating Spares Mixes with ISAAC

INTRODUCTION

In "requirements mode" ISAAC calculates the best spares mix to produce a user-specified target availability. The "evaluation mode" reverses that sequence — ISAAC calculates the availability rate provided by a user-specified spares mix. The evaluation mode is usually used to determine how well a spares mix performs under conditions that are different than the conditions of the original requirements run that produced the mix. The model uses a specified spares mix as the starting point (the spares in the inventory) for a model run. The evaluation function then provides the user with two major alternatives. The user can simply evaluate the spares mix under different conditions (such as different flying hours, a different day of the war, or different item data). ISAAC can also determine what spares should be added (optimally) to the mix if it does not meet a specified target. This chapter describes the features of the evaluation mode of operation.

The evaluation mode is only accessible in the complete version of ISAAC. It is not accessible in the limited version of ISAAC. The information in this chapter and any reference to ISAAC is only applicable to the complete version of ISAAC.

Special Purpose Evaluation for Assessments

One of ISAAC's main functions is to evaluate a spares mix as part of an overall logistics (spares, support equipment, maintenance, etc.) assessment of aircraft readiness. For that assessment, ISAAC estimates the aircraft availability yielded by a specific set of assets during a specific war scenario. The item data base should contain all the standard ISAAC data (demands, repair times, etc.) as well as the total assets (all the spares in the inventory — on hand, due in from maintenance, and due in from procurement within the evaluation time frame) entered in the initial asset field of ISAAC. To better evaluate wartime scenarios, ISAAC has a number of features such as evaluation of a spares mix for any day of the war, a **Multi-Day Evaluation** feature that allows the user to evaluate each day of the war, and a ***Critical* Item** list that identifies the items that are grounding the most aircraft on a specific day. We will describe each of those features in this chapter (except for the ***Critical* Item** list which is described in Chapter 5).

ISAAC can evaluate under both long range or short range conditions. For long range evaluations, the user-inputted data projects conditions for a future period of time (assets, demand rates, flying hours, procurement budgets, etc.) For short range evaluations, the user inputs data reflecting current conditions, particularly the current inventory assets. With the short term evaluation, ISAAC can evaluate aircraft performance for a war scenario that starts in the immediate future (say within the next few months). This time frame of a few months is used because ISAAC assumes steady state conditions at the start of the war and though ISAAC does incorporate current assets, it does not incorporate where those assets are (on hand or due in). For instance, ISAAC uses the average number of items in maintenance (i.e., repair pipelines) and the average number of backorders but not the actual values. In a few months, steady state conditions may be an appropriate assumption; but, it is not a valid assumption if the user wants to estimate aircraft availability today. To evaluate your current capabilities and determine what items are grounding aircraft today, you need to use databases that track those statistics not ISAAC.

Chapter Outline

Specifically, this chapter presents the following information:

- The model operation section presents the basic evaluation run sequence required to run all types of evaluations from the Evaluation Setup Screen.
- The implementation details describes how to select a spares mix, how to treat that spares mix (as buys or assets), whether or not to add to that mix to reach a target, how to perform a multi-day evaluation, and other parameter settings that impact the evaluation.
- A flying hour deceleration feature of ISAAC improves wartime demand estimates based on a recent LMI demand forecasting study. Though this feature is applicable to evaluation and requirement runs, we describe it in this section because it uses inputs similar to the multi-day evaluation.

Because this is only a demonstration model, the model will not be able to use data from previous ISAAC model versions but must create its own data by importing kit information.

Basic Evaluation Run Sequence

An evaluation run is similar to a requirements run at the start but then the user must perform a few additional steps. This sequence starts at the point where the Model Parameters Screen is open:

1. Select a previous run (**Prev Run)** or baseline kit **(Baseline)**.

2. Press the **Modify** button and enter a new run description.

3. Edit the parameters and scenario, if necessary. For a multi-day run you need to also enter the daily sortie duration and maximum sorties per day for each aircraft on the scenario screen. To use the deceleration option you need to click on the **Decelerate Hrs ..** check box on the Flying Hour Scenario Screen and enter the deceleration constant in the **Factor** text box.

4. Press the **Run Evaluation** button (instead of the **Run Requirement** button).

5. Answer the three additional evaluation questions on the Evaluation Setup Screen: what spares mix to use, how to treat that spares mix (as buys or assets), and whether to permit the model to buy additional spares, if necessary, to reach the target or run a strait evaluation. If you choose **No - Evaluate Selected Spares Mix Only** you will be given the choice of making a standard 1 or 2 day run or of running a multi-day evaluation over a consecutive series of days.

6. Press the **Continue** button once the answers are selected.

7. View (and if necessary edit) the selected spares mix.

8. Close the browse window of the spares mix.

9. Select the **OK** button after the model checks to see if the kit (selected in step 1) has similar NSNs to the selected spares mix (step 5 - top box of Figure 8-1).

Implementation Details

There are two possible starting points in the evaluation run mode (as in any ISAAC run). You may proceed from either an existing baseline kit (e.g., one that was just created via **Global Changes (Sensitivity) to Kit**) or a previous model run. Each of these points contains the requisite basic input component data and the parameters needed to make a model run. If necessary you can edit the parameter and component data. If you choose the previous run as the start point, there will be a spares mix associated with it. You will have the option of using this spares mix or selecting another one. If you choose to start from a baseline kit, there will be no spares mix associated directly with it and the model forces you to select one.

Once you select the kit and edit the parameters, you then press the **Run Evaluation** button and Figure 8-1 appears. The screen in Figure 8-1 allows the user a lot of flexibility on how to run the evaluation as we now describe.

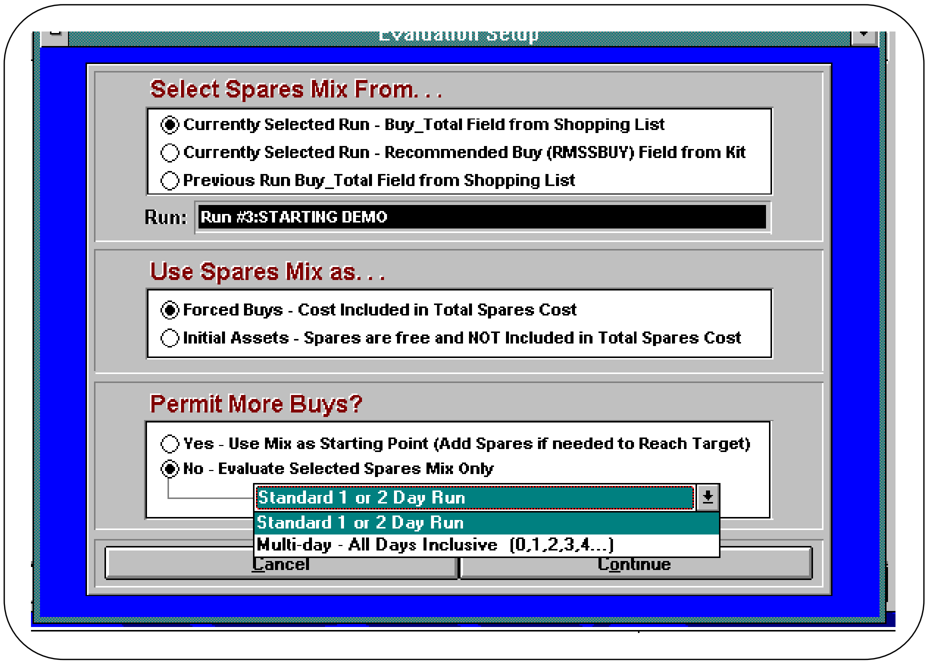


Figure 8-1
The Evaluation Setup Screen

Select Spares Mix

The user must first select the spares mix to evaluate. The user must select different analysis types each by pressing one of the following radio button options:

– **Currently Selected Run - Buy_Total Field from Shopping List**. When you select this analysis type, the model uses the spares mix, specifically the Buy Total value for each item, from the kit you selected at the very start of this process from the previous run list. Suppose you want to analyze how the Starting Demo spares mix, which estimated requirements for day 0 and day 24, performs on two intermediate days, day 5 and 15. You first select the Starting Demo, modify the parameters to reflect the proper analysis days of 5 and 15 (along with other changes such as the cannibalization option consistent for those days), and press the **Run Evaluation** button. Pick the radio button for the **Currently Selected Run - Buy_Total Field from Shopping List** option and ISAAC is ready to evaluate that mix on day 5 and 15.

– **Currently Selected Run - Recommended Buy (RMSSBUY) Field from Kit**. When you select this analysis type, the model uses a previously stored spares mix solution as the starting point for the evaluation. One of the fields in the kit is called RMSSBUY (the recommended manufacturer system stock buy quantity). With this evaluation setting, the user can estimate the availability and cost of purchasing what the manufacturer recommends. Though the field is set up for a standard initial procurement, where this information is assumed to be imported with the rest of the component level information, the user can use this field to specify any spares mix.

– **Previous Run Buy_Total Field from Shopping List**. With this analysis type selected, the model uses a spares mix from an alternate run and possibly an alternate kit. For instance, lets say you want to determine how well the spares mix from our Starting Demo performs if each item's demand rate increased by 10%. You would first create a copy of the Starting Demo kit and increase each component's demand rate by 10% (you can do this with the Copy Baseline Kit- Sensitivity Changes). Then select the newly created kit from the baseline kits when you run the model, enter the run description and other parameters, and press the **Run Evaluation** button (Figure 8-1 appears). By selecting the **Previous Run Buy_Total Field from Shopping List** radio button, you can than choose the Starting Demo spares mix even though it is from a different kit. Since this option allows you to pick any run, you have to make sure there are some NSN numbers in common or else the model will cancel the operation as we will describe soon. Also, if you originally selected a baseline kit to start the run, the model forces you to select a previous run since a baseline kit contains no spares mix.

Use Spares Mix

This group of options provides the user with flexibility in using the spares mix just selected. The user can either treat the spares mix as part of the total spares to be purchased or as "free" initial assets. For that the user selects one of the following radio button options displayed in the middle box of Figure 8-1, labeled "Use Spares Mix as..".

– **Forced Buy — Cost Included in Total Spares**. When you select this analysis type, the model includes the spares mix in the total spares cost (a budget estimate). In other words, you force the model to purchase the same spares at the same cost as the selected model run but you evaluate that spares mix under different conditions.

– **Initial Assets — Spares are free and NOT Included in Total Spares Cost**. When you select this analysis type, the model does *not* include the spares mix in the total spares cost (a budget estimate). In other words, the model treats the spares mix as initial assets that have been previously procured and are not included in the total cost.

Permit More Buys?

The user can decide whether to purchase additional spares over the selected spares quantities in order to meet the target. The user selects one of the following radio button options displayed in the bottom box of Figure 8-1, labeled "Permit More Buys".

- **Yes — Use Mix as Starting Point (Add Spares if needed to Reach Target)**. When you select this analysis type, the model adds to the selected spares mix, if needed, to achieve the availability or cost targets. Thus, the model can optimally add spares to the mix but will not subtract spares that are not needed to meet the target.

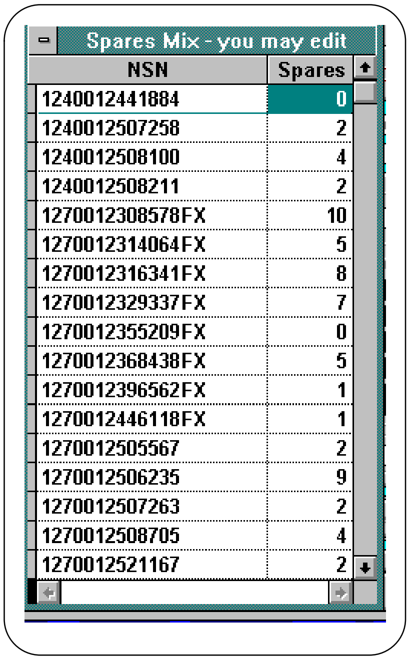
- **No — Evaluate Selected Spares Mix Only**. When you select this analysis type, the model uses only the selected spares mix specified — no more or less spares. Since the evaluation only evaluates a single spares mix, no curve table is produced. The advantage of this option over the alternative is that with this, ISAAC can evaluate extreme cases that contain very unbalanced mixes of spares: too few or too many spares for some items. For instance, if an item has a mean pipeline of 100 and a spares mix of only 5, the model can evaluate the large number of backorders (over 90) and the resulting impact on availability (grounding all of the fleet). When you select this option a drop-down list box is displayed with the following entries:

  - **Standard 1 or 2 Day Run**. This will result in a standard evaluation run of the model.

  - **Multi-day - All Days Inclusive (0,1,2,3,4 ...)**. This will result in an evaluation run of the model. The model will evaluate the availability that this kit yields on each consecutive day to be analyzed. The model will evaluate the kit from **Day 1** to **Day 2** inclusive. We further describe this option in the multi-day run section of this chapter.

In our earlier example, we started with the Starting Demo run that estimated requirements for day 0 and day 24. We then took that spares mix and evaluated it for days 5 and 15. If you selected **Yes — Use Mix as Starting Point**, the model adds spares to that initial mix until it reaches the targets. If you selected **No - Evaluate Selected Spares Only**, the model evaluates only that spares mix, even though it might not reach the target. (If days 5 and 15 are less demanding days than 0 and 24, then the model estimate of availability is greater (ENMCS lower) than the each analysis day's respective NMCS target and selecting either **Yes - Use Mix as Starting Point** or **No - Evaluate Selected Spares Only** will produce identical results.)

Once you have made your choice in each section of the Figure 8-1 screen, you press the **Continue** button (or **Cancel** to get completely out of the model run). You then see Figure 8-2, which is a list of the spares mix you selected with the first set of radio buttons (NSN and spares quantity). This allows you to make sure you selected the right mix of spares. This screen also lets you edit the spares quantity of each component. Close the window, by clicking the control menu box in the upper left corner of the window, to continue processing the evaluation.


Spares Mix - you may edit

NSN	Spares
1240012441884	0
1240012507258	2
1240012508100	4
1240012508211	2
1270012308578FX	10
1270012314064FX	5
1270012316341FX	8
1270012329337FX	7
1270012355209FX	0
1270012368438FX	5
1270012396562FX	1
1270012446118FX	1
1270012505567	2
1270012506235	9
1270012507263	2
1270012508705	4
1270012521167	2

Figure 8-2
The Spares Mix - you may edit spares quantities in this window

The final step of the evaluation pre-run checks is the comparison of the number of NSNs between the input kit and the spares mix as an error check. If you use the currently selected run, as the spares mix, the number of NSNs will always match; however, if you pick a previous run with a different kit number than the original kit, they may not match. Thus, the model performs a check to determine whether the kit and the spares mix have matching NSNs. The kit and mix NSNs do not have to be identical to each other; but, they must have at least some matching NSNs in common to continue processing the evaluation. For instance, if you made a run with a kit of 50 NSNs then created a new kit with 20 more NSNs added to the original 50 NSNs making 70 NSNs. If you make a run with the new kit using the spares mix of the old kit, only 50 NSNs will have a spares level. The model automatically sets the spares level of the other non-matching 20 NSNs to zero.

Once the pre-run checks are completed ISAAC will run the evaluation. The ASM Window will be displayed and the model will print to the screen as it processed the data. From this point on the processing of the model is identical to the requirements run processing.

Once the evaluation is completed, the model stores descriptive information about the spares mix you selected in the comment field of the parameter. For instance, if you selected the starting demo for your spares mix, the comment field will read "Evaluation - Spares from Run #3: STARTING DEMO".

Multi-day Evaluations

The Multi-day Evaluation feature of ISAAC evaluates aircraft availability on each day of the war in one ISAAC pass (i.e., the model evaluates a particular spares mix to determine the availability on days 0, 1, 2, ... to the end of the war). We use the term multi-day evaluation to refer to consecutive daily evaluations for periods of time that are greater than two days. This option is similar to the standard 2 day evaluation (**Evaluate Spares Mix Only - 1 or 2 Day Standard Run**) with several additional advantages. It automates the equivalent of making many two day model runs so that a single model run can evaluate each day of the war. It also allows the current day activity to be impacted by the results of previous days activity. That is the model bases the flying hours of a particular day on the number of available aircraft (aircraft minus the NMCS from the day before) and reduces the users specified flying hours if there are not enough aircraft to fly all the sorties, as we now describe.

Let's assume that over the course of a war, you expect that 10 aircraft will each fly 1 sortie a day and each sortie lasts 2 hours (what we term the **Flying Hrs/Sortie**). The total ISAAC flying hours (entered in the Flying Hour Scenario Screen) is then equal to 20 for each day of the war (10 aircraft times 1 sortie / day times 2 hours per sortie). Now lets assume you are evaluating a kit with a multi-day run and the available planes drop below 10 to 9 on day 15. That means on day 16, you start the day with only 9 aircraft so you need to reduce flying hours from 20 to 18 (9 aircraft times 1 sortie / day times 2 hours per sortie).

Sometimes an air force might decide to fly two sorties per day for one aircraft on day 16 to bring the planned flying hours back up to 20. For that case, ISAAC also allows you to enter the maximum number of sorties an aircraft can fly per day (what we term the **Max Sorties/day**). Thus, even though the air force plans on one sortie per day for each aircraft, if needed the aircraft can fly up to the maximum sortie rate. Continuing our simple example (but now considering **Max Sorties/Day**), the number of available aircraft could drop as low as 5 and that would keep the flying hours at 20 [5 aircraft times 2 (**Max Sorties/Day**) times 2 **Flying Hrs/Sortie**], but if the number of available aircraft drops to 4 then that reduces the flying hours to 16 [4 times 2 (**Max Sorties/Day**) times 2 **Flying Hrs/Sortie**]. In this later case, we term the 20 hours the *planned* flying hours (PFH) and the 16 hours the *flown* flying hours (FFH).

Similar to the example, ISAAC limits the flying hours to no more than the MCS times the **Max Sorties/Day** times the **Flying Hrs/Sortie** per aircraft. ISAAC looks at the whole PDF (Probability Distribution Function) of MCS and reduces flying hours by the portion of that PDF that falls in the limiting area as describe by the following equation.

$$FFH = \sum_{MCS} p(MCS) \times Min(PFH, (MCS \times MaxSorties/Day \times FlyingHrs/Sortie))$$

where

p(MCS) is the probability of MCS being available on the previous day.

MaxSorties / Day is the maximum number of sorties per day.

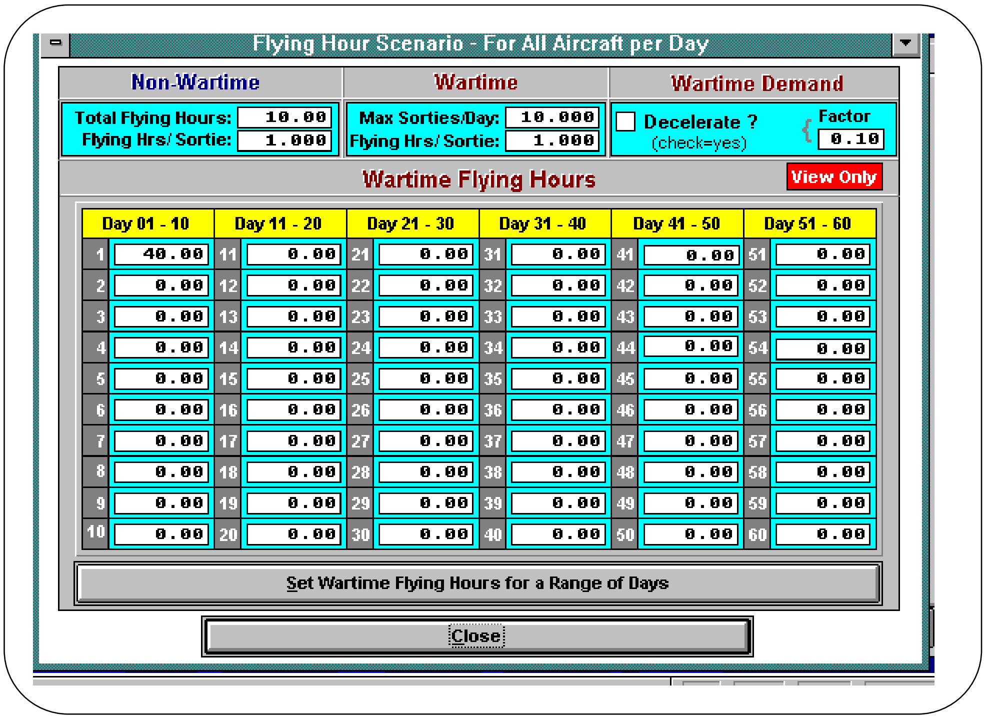


Figure 8-3
Flying Hour Scenario — For All Aircraft per Day

The user enters the wartime sortie duration (**Flying Hrs/Sortie)** and the maximum sorties per day (**Max Sorties/Day)** in the Flying Hour Scenario Screen (Figure 8-3). The model assumes they are constant for the entire war; though the total flying hours (a user input) and number of sorties per day may vary. The model then calculates the sorties planned, sorties flown, availability planned, availability flown, flying hours planned, flying hours flown for each day of the war, as we will describe in the browse report section.

IMPLEMENTATION

As described earlier, this option is accessed via the evaluation screen (Figure 8-1). For the option Permit More Buys? — You must select **No — Evaluate Selected Spares Mix Only**. (When you select this analysis type, the model uses only the selected spares mix specified — no more or less spares.) This will open a drop-down list box with two selections. Select **Multi-day - All Days Inclusive (0,1,2,3,4 ...)**. This will result in an evaluation run of the model. The model will evaluate the availability that this kit yields on each consecutive day to be analyzed. The model will evaluate the kit on each day from day 0 through the **1st**

Analysis Day or **2nd Analysis Day (if applicable)**. (If non-wartime flying hours are zero the first analysis day will be day 1).

Once the model has completed the evaluation run, the Performance Report Window will be displayed. The Performance Report Window will display ENMCS, EBO and Confidence data in the Day 1 values only. These values pertain to the last day of the evaluation (except for Buycost which is the total spares buy cost and applies across all days that this kit has been accessed against). To see ENMCS, EBO and Confidence data for any or all of the other accessed days you will need to browse the Multi-day Evaluation Window, which we describe in the next section.

Multi-day evaluation runs do not produce curve information. All of the other input, output and miscellaneous browse windows can be opened and viewed for a multi-day evaluation run. The information viewed in these browse windows only pertains to the specific analysis days the user entered in the **1st Analysis Day** and **2nd Analysis Day** fields of the Model Parameters Screen.

MULTI-DAY BROWSE

Multi-Day Evaluation can only be selected through the **View Output** pull-down menu from the main menu. This selection allows you to browse the multiday evaluation data. This multi-day evaluation data will pertain to the model run description displayed on the Model Parameters Screen. Clicking on the **Multi-Day Evaluation** menu item will open the Choose Multi-Day Fields pick-list (Figure 8-4).

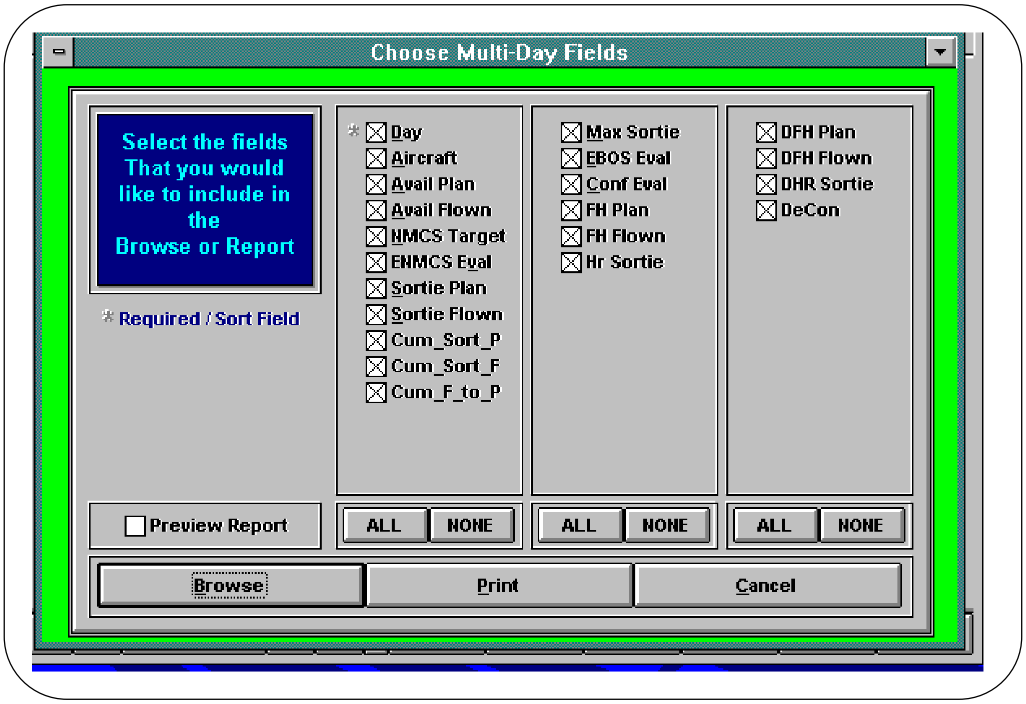


Figure 8-4
Choose Multi-Day Fields Picklist

The **Multi-Day Evaluation** Window (Figure 8-5) is a modified browse window indexed only on the day of the evaluation. ISAAC creates a temporary database file when you browse **Multi-Day Evaluation** so that you can export a single model run's data to other software for further analysis. The separate dbf file is in the IAF_PORT subdirectory and is called out_mult.dbf. The temporary files are destroyed if you view the same browse again.

MultiDay - Model #30: IMPTFT5: MULTI-DAY EVAL OF DEMO

Day	Aircraft	Avail_plan	Avail_flow	Enmcs_eval	Sort_plan	Sort_flown	Cum_sort_f	Cum_f_to_p
0	20	70.00	70.45	5.91	10.00	10.00	10.00	1.00
1	20	70.00	84.65	3.07	50.00	42.26	52.26	0.87
2	20	70.00	82.15	3.57	50.00	50.00	102.26	0.93
3	20	70.00	79.60	4.08	50.00	49.28	151.54	0.95
4	20	70.00	78.40	4.32	50.00	47.76	199.30	0.95
5	20	70.00	77.70	4.46	50.00	47.03	246.33	0.95
6	20	70.00	76.90	4.62	50.00	46.61	292.94	0.94
7	20	70.00	76.05	4.79	50.00	46.13	339.06	0.94
8	20	70.00	75.15	4.97	50.00	45.62	384.68	0.94
9	20	70.00	74.20	5.16	50.00	45.10	429.78	0.93
10	20	70.00	73.35	5.33	50.00	44.53	474.31	0.93
11	20	70.00	72.40	5.52	50.00	44.00	518.31	0.93
12	20	70.00	71.50	5.70	50.00	43.45	561.76	0.92
13	20	70.00	70.65	5.87	50.00	42.90	604.66	0.92
14	20	70.00	69.75	6.05	50.00	42.38	647.04	0.91
15	20	70.00	69.00	6.20	50.00	41.86	688.89	0.91
16	20	70.00	68.20	6.36	50.00	41.39	730.28	0.90
17	20	70.00	67.40	6.52	50.00	40.93	771.21	0.90
18	20	70.00	66.55	6.69	50.00	40.45	811.65	0.89
19	20	70.00	65.70	6.86	50.00	39.94	851.59	0.89
20	20	70.00	64.85	7.03	50.00	39.42	891.01	0.88

Figure 8-5
The Multi-Day Evaluation window

For the multi-day browse field name, the suffix "plan" means the user plans for this (i.e., a user input) and the suffix "flown" means the model output. In our previous example, we termed the 20 hours the user entered as a scenario parameter as the *planned* flying hours versus the 16 hours the model predicted were actually *flown* by the available aircraft. The multi-day browse field definitions are as follows:

Day — The day of analysis with 0 representing peacetime and all other numbers representing the respective day of the war.

Aircraft — The total number of aircraft the user enters on the parameter screen.

Avail Plan — (1 - NMCS Target) / Aircraft expressed as a percentage.

Avail Flown — (1 - ENMCS Eval / Aircraft expressed as a percentage.

NMCS Target — The NMCS target from the Model Parameters screen. All days up through the **1st Analysis day** use the **1st NMCS Target**, otherwise the **2nd NMCS Target** is used.

ENMCS Eval — The model calculated expected not mission capable for supply.

Sortie Plan — (FH Plan / Hr Sortie)

Sortie Flown — (FH Flown / Hr Sortie)

Cum_Sort_P — The cumulative number of sorties planned as of the particular Day. For instance, on day 20 , the cumulative sorties planned equals the sum of all sorties planned from day 0 through day 20.

Cum_Sort_F — The cumulative number of sorties flown as of the particular Day. For instance, on day 20 , the cumulative sorties flown equals the sum of all sorties flown from day 0 through day 20.

Cum_F_to_P — The ratio of the number of flown sorties to planned sorties (Cum_Sort_F) / (Cum_Sort_P) as of the particular Day.

Max Sortie — Wartime **Max Sorties/Day:** The maximum number of sorties an aircraft can fly per day (user input enter in the Flying Hour Scenario screen).

EBOS Eval — The calculated expected back orders, by day.

Conf Eval — The calculated confidence level of meeting the NMCS target, by day.

FH Plan — The planned flying hours. This is the flying hours from the Flying Hour Scenario screen, by day.

FH Flown — The minimum of the planned flying hours and (Aircraft - ENMCS Eval of previous day) * (Hr Sortie) * (Max Sortie).

DFH Plan — is the decelerated flying hour plan (see Improving Forecast of Wartime Demand section).

DFH Flown — is the decelerated flying hours flown (see Improving Forecast of Wartime Demand section).

Hr Sortie — This is the number of hours per sortie. The values are taken from the non-wartime (for day 0) and Wartime **Flying Hours / Sortie:** fields from the Scenario screen.

DHR Sortie — is the decelerated number of hours per sortie. In our example the 1.1 is the decelerate sortie duration when the non-war and the wartime sortie duration is 1 and 2 respectively, and the slope is 0.1. If the non-war flying hour duration is not exactly one or the wartime duration is not a whole number, the model uses a proportional formula to estimate the deceleration duration.

DeCon —is the deceleration conversion or the ratio of the decelerated to the actual sortie duration (see Improving Forecast of Wartime Demand section).

Stock Option Impacts on Evaluation

The Stock Option parameters of **Include Starting Assets?** and **Use Pre-specified Buy Quantity?** (accessed via the **Stock / Resupply / Other Options** button) affect ISAAC evaluation run output. Every kit has values for both of those parameters set as well as the item data associated with those parameters. In general the Stock Option parameter selections are used as described in Chapter 3; however, sometimes with an evaluation the meaning is slightly different as we now describe:

STARTING ASSETS

Include Starting Assets? With this drop-down list box selection, the user can determine whether to add the selected spares mix to the existing kit assets or not. The main difference when running an evaluation run occurs when the user selects Use Spares Mix as initial assets. For that case, if the parameter setting is - **Do Not Use Starting Assets** - the model ignores any kit starting assets (as usual) but now inserts the output spares mix from the selected model run as initial assets. If the parameter setting is -**Use Assets: InitAsset + FreeAsset** - the model adds the spares mix to the kit's initial assets. Thus when you view the Shopping List or Input-Output browse windows you see the initial assets that include the selected spares mix; however, when you view the Component Kit initial assets (i.e., the input value) the quantity you see is the initial asset quantity without the spares mix. In other words, the selected spares mix is treated as model output and can only be viewed via the output browse windows (Shopping List and Input-Output). As always, the model does can not change the input kit data so that previous model runs that used this data remain consistent.

Table 8-1 displays a simple example for an item and the different combinations of stock and evaluation options. The input includes the user selection of how to use the spares mix and the actual item spares quantity. Other inputs include the parameter stock option setting for assets and the kit initial asset value. Given those 4 inputs, the table shows the possible permutations of the input field values and the corresponding output field values — providing example values to the concepts we described earlier. The outputs are the initial assets (adjusted to reflect the evaluation option if needed), the spares buy, and the spares target. Notice, that the model treats that spares mix as either the initial assets or the spares buy quantity; it does not treat the spares mix as the spares target unless there are no assets.

Table 8-1.
Item Evaluation Output under Different Input Settings

Input				Output		
Evaluation input		Parameter input	Kit input	Shopping and input-output window browses		
Option use spare mix as	Selected spares mix item value	Parameters: include starting assets?	Item initial asset value	Initial assets	Spares buy	Spares target
Initial Assets	5	Yes	4	9	0	9
Initial Assets	5	No	4	5	0	5
Forced Buys	5	Yes	4	4	5	9
Forced Buys	5	No	4	0	5	5

PRESPECIFIED BUY QUANTITIES

Use Pre-specified Buy Quantity? In the standard requirements run, the user can use this parameter option to force the model to buy at least the specified spares level if the user selects **Yes — Buy Quantity = Items's Neglv**. When you run an evaluation, the model ignores the parameters setting for pre-specified buys and will not buy the minimum quantity specified in the items negotiated level field.

Also, in the standard requirements run the model will automatically buy the specified spares quantity if the item NOP field has a value of FIX, NOP, or CAP. In general this is also the case for an evaluation run. The only exception is when the user selects the evaluation radio buttons: **Forced Buys Cost Included in Total Spares Cost** and **Yes — Use the Mix as a Starting Point (Add Spares if needed to Reach Target)**. In that case, the model overrides the Negotiated Level (Neglv) spares quantity in the item field. The reason for that exception is that the user is really telling ISAAC to do contradictory things so ISAAC chooses to perform the evaluation instructions. For instance, lets say you enter a FIX quantity of 4 for the negotiated level field and then ran an evaluation that says the spares mix for that item is 6. The model then is asked to specifically buy simultaneously 4 and 6 spares for this same item, which it can not do, so it assumes the evaluation value of 6 spares is correct.

IMPROVING FORECAST OF WARTIME DEMAND

ISAAC has a flying hour deceleration feature that improves wartime demand estimates. Usually, the air force uses non-wartime demand rates to estimate wartime demand rates. During a recent study, LMI has examined various factors that impact estimating wartime demand rates using peacetime demand data. The most important finding of the study (see article in the May 1996 USAF Journal of Logistics by Mike Slay, Dave Peterson and Craig Sherbrooke) is that demand is much more closely related to the number of sorties than the number of flying hours. We came to that conclusion after examining hundreds of thousands of sorties flown by every major aircraft type in the USAF.

Specifically, we estimate that the demand rate generated by a one hour sortie, increases by about 10 percent for each additional hour of sortie duration (what we term a 10% slope or a 0.10 deceleration factor in the scenario screen) for fighter aircraft and increases by 20% for transports and bombers (a 20% slope).

How does ISAAC incorporate this information? Lets say, that a hypothetical IAF non-wartime training mission for tactical aircraft averages approximately one hour whereas the wartime mission average is expected to be about 2 hours. Under a flying hour based demand assumption (current IAF assumption), the wartime missions would be expected to produce 2 times as many demand-based failures. Using our study results which forecast a 10% slope, the wartime missions would be expected to produce 1.1 times as many demand-based failures as the peacetime missions.

Since ISAAC uses demands per flying hour to project wartime failures, we decided to adjust the flying hour input to the model. Lets continue with our example of 1 hour peacetime and 2 hour wartime sortie duration. Suppose also that the wartime flying program is 80 hours per day (e.g., 20 aircraft flying, times 2 sorties a day / aircraft, and times 2 flying hours per sortie). However, since demands from a 2 hour sortie are only 10% more than from a 1 hour sortie, we would use 44 hours as the wartime flying program (e.g., 20 aircraft flying, times 2 sorties a day / aircraft, and times 1.1 flying hours per sortie) in the model instead of 80 hours. Thus, the model can correct for the decelerated impact of additional hours in a sortie by adjusting the sortie length and total flying hours. The model, which computes demands on a strict per-flying-hour basis, is "tricked" into correctly forecasting demands for the longer sortie. In our example, we term the 1.1 flying hours the decelerated number of hours per sortie **(DHR Sortie)** and the 44 hours as the decelerated flying hour plan (**DFH Plan**), and the ratio of the decelerated hours per sortie to the actual hours per sortie the deceleration conversion (**Decon)**. Those three variables are all presented in the **Multi-Day Evaluation** Window (see Figure 8-5).

Glossary of Screen Terms and Field Names

This is a brief description of each of the descriptors presented on the following ISAAC screens and Browse windows.

– Screens: [i.e., (Kit) Parameters; (Model) Parameters; (Kit) Component Data; (Kit and Model) Pipeline, Resupply & Options; (Kit and Model) Flying Hour Scenario, and Evaluation Setup] the Define Kit ID dialog box, the Sensitivity Change page frame, and the Performance Report Window. Screens are those objects within ISAAC where a user can edit data.

– Browse windows [i.e., (Model) Component Data; (Model) Run Log; Shopping List; Pipeline Data; Curve; Critical Item List; Multi-Day Evaluation; Yearly Cost; View Input-Output; and Statistics windows). Browse windows are those objects within ISAAC where a user can modify the presentation of data but not edit the data itself.

In addition to the description, the following information is provided for each field name: field size and type, input and output locations (i.e., browse windows), where the data can be edited (if applicable) and cross reference information with Appendix A. There are some differences between the Kit and Model Parameters Screens. Therefore these screens are referenced explicitly in the text. There is no difference between the Kit and Model Pipeline, Resupply, & Options Screens. Therefore when we will refer to the "Pipeline, Resupply, & Options Screen" in the text we are indicating that the field can be edited in both the Kit and Model screens.

% Confidence — Definition: This is the probability of meeting the NMCS target, using the given Day 1 + Day 2 Cost, expressed as a percent. Field: 8 character spaces. Output location: Curve Window

% Availability — Definition: The percent of available aircraft equals the total number of aircraft under consideration minus the ENMCS aircraft with the result divided by the total number of aircraft under consideration. Field: 8 character spaces. Output location: Curve Window

1st Analysis Day Information — Consists of the **1st Analysis Day:**, **1st NMCS Target:**, **1st Confidence:**, **1st Budget:**, and **Cannibalization (Thru 1st day):** fields.

1st Budget: — Definition: This is the budget constraint that is to be applied to the **1st Analysis Day:**. An optional input, it is mutually exclusive of the **1st Confidence: %** constraint (i.e., you may enter a confidence % or a budget target but not both). In the budget constrained mode, the model may not exceed the NMCS target on either or both days but it will not exceed the budget constraint. Field: Text box with 11 numeric spaces (2 decimal places). Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Curve_val2]

1st Confidence: — Definition: This is the probability, expressed as a percentage, of meeting the NMCS target. The model optimizes the probability that the number NMCS is not greater than the target. This is an optional input and is mutually exclusive of the **1st Budget:** constraint (i.e., you may enter a confidence % or a budget target but not both). Field: Text box with 3 numeric spaces. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Confidnc1]

1st Analysis Day: — Definition: The first day to be analyzed. A "0" for the model's analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A "24" (or entering any day from 1 through 99 of the wartime scenario) translates into having the model procure spares for the 24th day of the wartime scenario that starts at the end of the coverage period. Field: Text box with 2 character spaces. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Day1]

1st NMCS Target: — Definition: This is the not mission capable for supply target or the allowable number of aircraft on ground (AOG) specified by the user for the first analysis day. The model minimizes the spares cost required to meet the target or — if constrained by the budget — purchases the spares that yield the best spares mix without exceeding the budget constraint. The NMCS target serves two functions: it acts as a curve target to stop the optimization process once the target is reached and it identifies the number of aircraft that are likely candidates for cannibalization according to the maintenance assumption specified. Field: Text box with 5 character spaces. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Nmcs1]

2nd Analysis Day: — Definition: The second day to be analyzed. A "0" for the model's analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A "24" (or entering any day from 1 through 99 of the wartime scenario) translates into having the model procure spares to cover failures from the first to be analyzed up to the 24th day of the wartime scenario (that starts at the end of the coverage period). For the second analysis day, the model considers those spares bought on the first day as being in the inventory. It then purchases additional spares to meet the specified second-day target or second-day incremental budget constraint (if applicable). Thus, the spares that the model selects for the second day make up the difference between the previous days' buy and those required to meet the second target. A blank (shown as an "*") indicates that no second-day analysis will be performed. Field: Text box with 2 character spaces. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Day2]

2nd Budget: — Definition: This is the incremental budget constraint that is to be applied to the **2nd Analysis Day:**. The total combined budget equals the value entered here plus the value entered for the **1st Analysis Day:**. Field: Text box with 11 numeric spaces (2 decimal places). Input locations: Kit Parameters Screen; Model Parameters Screen. [cross reference with Curve_val2]

2nd Confidence: — Definition: Same as for day 1. Field: Text box with 3 numeric spaces. Input locations: Kit Parameters Screen and Model Parameters Screen. [cross reference with Confidnc2]

2nd NMCS Target: — Definition: This is the not mission capable for supply target or allowable number of aircraft on ground (AOG) specified by the user for the **2nd Analysis Day:**. The model purchases enough spares to meet that target or — if constrained by the budget — purchases the spares that yield the best spares mix without exceeding the budget constraint. Field: Text box with 5 character spaces. Input locations: Kit Parameters Screen; Model Parameters Screen. [cross reference with Nmcs2]

2nd Analysis Day Information — Consists of the **2nd Analysis Day:**, **2nd NMCS Target:**, **2nd Confidence:**, **2nd Budget:**, and **Cannibalization (Thru 2nd day):** fields.

Ac_deliver — Definition: This is the year that the aircraft will be delivered. It is the start point of the spares coverage period and the end point of the budget period. Field: 7 numeric spaces (2 decimal places). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **Aircraft Delivery Year**]

Achieved Confidence of NMCS Target — Definition: The achieved probability of meeting the NMCS target expressed as a percent. Field: numeric. Output location: Performance Report Window. [cross reference with Confidence1 and Confidence2]

Add button — Definition: Selecting this button creates a new component record using the default component fields. Input location: Kit Component Data Screen

Aircraft **Delivery Year:** — Definition: This is the calendar year in which the aircraft will be delivered. The model uses the first day of this year as the end of the budget period and the beginning of the spares coverage period. Field: Text box with 7 numeric spaces (2 decimal places). Input locations: Kit Parameters Screen; Model Parameters Screen. [cross reference with Ac_deliver]

Aircraft **Number:** — Definition: The number of weapon systems per base. Field: Text box with 3 character spaces. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Nunits]

Aircraft — Definition: The total number of aircraft the user enters in the **Aircraft Number:** field of the Parameter Screen. Field: 11 numeric spaces. Output location: Multi-Day Evaluation Window. [cross reference with **Aircraft Number**]

Apply button — Definition: Selecting the **Apply** button will replace the field values in this kit database with the values contained in the **NewValue*** column. On the screen, it will replace the original [field name] values which are displayed in the column to the left (the field header will contain the name of the ISAAC field) with the values contained in the **NewValue*** column. To the left of the original [field name] is the **Nsn** field. These records are sorted in NSN order. The **Nsn** field column is a reference point to show you which record the new and current field values correspond to. Input location: Preview / Edit / Apply page of Sensitivity Change page frame.

Apply Change To: — Definition: This is a drop-down list box consisting of **All** and **Filter**. This field determines whether the changes will be applied globally (**All**) or to a user specified subset (**Filter**) of item records. Input location: Define Changes page of Sensitivity Change page frame.

As_kitdesc — Definition: The Kit Description field displays the information entered in the **Description** field of the Define Kit ID dialog box. This information is entered by the user during the kit creating process. Field: 20 character spaces. Input location: Component Data Window. [cross reference with **Kit Description**]

As_kitid — Definition: The kit ID number is a unique number assigned by ISAAC to each baseline kit for tracking and identification purposes in the library of available kits. Field: 6 character spaces. Input location: Component Data Window. [cross reference with Kit #]

As_kitname — Definition: The kit name is a name associated with a particular kit. This name is entered by the user during the kit creating process. Field: 15 character spaces. Input location: Component Data Window. [cross reference with **Kit Name**]

Avail_flow[n] — Definition: This is the expected aircraft availability for the specified day of the multi-day evaluation. It is (1- ENMCS / Number of Aircraft) expressed as a percentage. Field: 11 numeric spaces (2 decimal places). Output location: Multi-Day Evaluation Window.

Avail_plan — Definition: This is the target aircraft availability for the specified day of the multi-day evaluation. It is (1- NMCS target / Number of Aircraft) expressed as a percentage. Field: 11 numeric spaces (2 decimal places). Output location: Multi-Day Evaluation Window.

Awp — Definition: Awaiting parts is the expected number of this component that are awaiting parts from any of their sub-components. This field is restricted to components that have lower indentured components (i.e., that have one or more SRUs). (The AWP for SRUs on the lowest indenture level is 0.00)Field: 7 numeric spaces (2 decimal places). Output location: Pipeline Data Window

Awp_1 — Definition: Awaiting parts is the expected number of this component that are awaiting parts from any of their sub-components on the first day to be analyzed. The Awp_1 for SRUs on the lowest indenture level is 0.00. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Awp_2 — Definition: Awaiting parts is the expected number of this component that are awaiting parts from any of their sub-components on the second day to be analyzed. The Awp_2 for SRUs on the lowest indenture level is 0.00. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

AWP — Definition: Awaiting parts is the expected number of all components that are awaiting parts from any of their sub-components. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Average**, **Minimum**, and **Maximum:** 7 numeric spaces (2 decimal places) each; **Sum**: 10 numeric spaces (2 decimal places). Output location: Statistics Window

Base Repair Time — Definition: Base repair time in days for this component. This is the number of days from item failure through the repair of the item to a serviceable status. Field: Text box with 6 numeric spaces. Input location: Kit Component Data Screen [cross reference with Ibrtp and Ibrtw]

Baseline button — Definition: Selecting this button opens the Available Kits Window that will enable you to select a new baseline of input parameters. A baseline consists of a kit and all associated input parameters including defaults for the Model Parameters Screen. The baseline selection allows the user to start from or return to a set starting point. Input location: Model Parameters Screen

Baspipe — Definition: Baspipe is the base pipeline mean. It is the number of components expected to be repaired at the base during the model run timeframe. The base repair time for each component during non-war (Ibrtp) and wartime (Ibrtw) conditions are inputs to the model. Field: 7 numeric spaces (2 decimal places). Output location: Pipeline Data Window

Baspipe_1 — Definition: Baspipe_1 is a base pipeline mean. It is the number of components expected to be repaired at the base on the first analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item window

Baspipe_2 — Definition: Baspipe_2 is a base pipeline mean. It is the number of components expected to be repaired at the base on the second analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

BASPIPE — Definition: Base pipeline is the number of all components expected to be repaired at the base during the model run timeframe. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Average**, **Minimum**, and **Maximum:** 7 numeric spaces (2 decimal places) each; **Sum**: 10 numeric spaces (2 decimal places). Output location: Statistics Window

BNRTSP — Definition: The non-wartime NRTS rate is an estimate of the percentage of the base demand that is not repaired at the base during non-wartime for all items. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Average**, **Minimum**, and **Maximum:** 8 numeric spaces (5 decimal places) each; **Sum**: 10 numeric spaces (5 decimal places). Output location: Statistics Window

Bnrtsp — Definition: The non-wartime NRTS (not reparable at this station) rate is an estimate of the percentage of the base demand that is not repaired at the base (that is the percentage that is sent to the depot for repair or condemned) during non-wartime. To estimate the NRTS rate, you must split base demand into 3 components: 1) what percentage of the total base demand is repaired at the base, 2) what percentage of the base demand is repaired at the depot, and 3) what percentage of the base demand is condemned at the base or depot. The sum of those three equals 100%. Thus, the NRTS rate equals 100 minus the base repair percentage or the NRTS rate equals the condemnation percentage plus the depot repair percentage. If the item is only repaired at the base (i.e., there is no depot repair of the item) then the NRTS rate equals the condemnation percentage. Field: 8 numeric spaces (5 decimal places). Input location: Component Data window; Location to edit information: Kit Component Data Screen. [cross reference with **Not Reparable at Station**]

Bnrtsw — Definition: Percentage of base not repairable this station (NRTS) wartime demands that are either condemned or sent to the depot for repair (overhaul) for this component. To estimate the NRTS rate, you must split base demand into 3 components: 1) what percentage of the total base demand is repaired at the base, 2) what percentage of the base demand is repaired at the depot, and 3) what percentage of the base demand is condemned at the base or depot. The sum of those three equals 100%. Thus, the NRTS rate equals 100 minus the base repair percentage or the NRTS rate equals the condemnation percentage plus the depot repair percentage. If the item is only repaired at the base (i.e., there is no depot repair of the item) then the NRTS rate equals the condemnation percentage. Field: 8 numeric spaces (5 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Not Reparable at Station**

Br_rrlru — Definition: the day that base repair starts for RR LRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Base Repair Begins**]

Br_rrrlru — Definition: the day that base repair starts for RRR LRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Base Repair Begins**]

Br_sru — Definition: the day that base repair starts for SRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Base Repair Begins**]

Brt — Base Repair Time (see Ibrtp & Ibrtw)

Budcode — Definition: (see Ibudcode) A user defined budget code from 1 to 99 which is used to record composite output parameters for a group of items (such as subsystems, LRUs, SRUs, Consumables etc.,) Field: 7 numeric spaces (index). Input location: Component Data Window; Output location: Critical Item, Performance Report, and Shopping List Data Windows. Location to edit information: Kit Component Data Screen. [cross reference with **Budget Code**]

Budget Code: — Definition: A user defined budget code from 1 to 99 which permits cost and availability subtotals to be generated by budget code.Ibudcode is envisioned to be used to identify all items that are part of a given subsystem (e.g., fire control , landing gear, etc.) so as to be able to predict subsystem parameters. Field: Text box with 2 numeric spaces. Input location: Kit Component Data Screen [cross reference with Ibudcode]

Budget Yr 1® 4 — Definition: These are the picklist fields for the Shop window that represent the calendar years looking back in time from the aircraft delivery year (e.g., if the aircraft delivery year is 1998 then Budget Yr 1 will be 1997 and Budget Yr 4 will be 1994). The values contained in Budget Yr 1→ 4 are the number of spares ordered in the respective calendar year at the component level. Field: 10 numeric spaces. Output location: Shopping List Data Window

Budget — Definition: This is either the dollar value specified in Cost_sum (rounded to the nearest dollar) or else it is the user value entered to constrain the annual budget. If the user keys in **F10** at the "Spares Cost Summary a PLTT before Spares Delivery" Screen without changing any of the budget values, this field will contain the dollar value specified in Cost_sum (rounded to the nearest dollar). If the user enters a budget value at the "Spares Cost Summary a PLTT before Spares Delivery" Screen and then keys in **F10** this field will contain that budget value. Field: 9 numeric spaces. Output location: Yearly Cost Window

Buy Cost — Definition: The purchase cost of the required spares broken out by analysis day. Field: 11 numeric spaces. Output location: Performance Report Window [cross reference with Buycost1 and Buycost2]

Buy_cost — Definition: The Buy_cost is the product of the Buy_total and item cost fields at the component level. This is the combined cost of the total number of each respective purchased component. Field: 12 numeric spaces (2 decimal places). Output locations: Shopping List Data and View Input-Output Windows

Buy_Cost1 — Definition: Buy_Cost1 is the combined cost of the total number of each respective purchased component in the first model run of the comparison. Buy_Cost1 is the product of the Buy_Tot1 and item cost fields (from Component Data). Field: 12 numeric spaces (2 decimal places). Output location: Shop Comparison Window

Buy_cost1 — Definition: Buy_cost1 is a sum of the products of the Buy_total and item cost fields (from Component Data) for all components with the same Budcode for the first analysis day. This is the combined cost of the total number of each respective purchased component. Consists of one distinct parameter for each budcode and a SUM. Field: 11 numeric spaces. Output location: Performance Report Window. [cross reference with Buy Cost]

Buy_Cost2 — Definition: Buy_Cost2 is the combined cost of the total number of each respective purchased component in the second model run of the comparison. Buy_Cost2 is the product of the Buy_Tot2 and item cost fields (from Component

Data). Field: 12 numeric spaces (2 decimal places). Output location: Shop Comparison Window

Buy_cost2 — Definition: Buy_cost2 is a sum of the products of the Buy_total and item cost fields (from Component Data) for all components with the same Budcode for the second analysis day. This is the combined cost of the total number of each respective purchased component. Consists of one distinct parameter for each budcode and a SUM. Field: 11 numeric spaces. Output location: Performance Report Window. [cross reference with Buy Cost]

Buy_day1 — Definition: The total number of spares purchased for the first analysis day, by component. Field: 10 numeric spaces. Output locations: Shopping List Data and View Input-Output Windows.

Buy_day2 — Definition: The total number of spares purchased for the second analysis day, by component. Field: 10 numeric spaces. Output locations: Shopping List Data and View Input-Output Windows.

Buy_Tot1 — Definition: The total of Buy_day1 and Buy_day2 by component for the first model run of the comparison. This is the total spares quantity that the model recommends purchasing by component. Field: 10 numeric spaces. Output location: Shop Comparison Window

Buy_Tot2 — Definition: The total of Buy_day1 and Buy_day2 by component for the second model run of the comparison. This is the total spares quantity that the model recommends purchasing by component. Field: 10 numeric spaces. Output location: Shop Comparison Window

Buy_total — Definition: The total of Buy_day1 and Buy_day2 by component. This is the total spares quantity that the model recommends purchasing by component. Field: 10 numeric spaces. Output locations: Critical Item, Shopping List Data and View Input-Output Windows.

Buycover — Definition: This is the abbreviated field name for the spares coverage period — the period of time during which the initial provisioning system is responsible for spares procurement for this weapon system. ISAAC estimates the spares required for that entire time period. After the coverage period, the model assumes the standard replenishment system can provide spares support. Field: 6 numeric spaces (2 decimal places). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **Spares Coverage Period**]

Buypeak — Definition: Specifies what pipeline values are to be bought sacrosanct to the level specified by LRU Percentage (on first and second days) and SRU Percentage (on first and second days). A value of ‘F’ indicates that the pipelines on the last day of the scenario will be used. A value of ‘T’ instructs the model to use the largest ‘peak’ pipeline over the scenario for each item. When “T” is selected the model computes the pipeline through each day to be analyzed for each item. The model purchases the pipeline for each item sacrosanct prior to conducting the marginal analysis for the kit. (The day on which each item attains its peak pipeline differs on the basis of the respective item's resupply/repair parameters.) Optionally, a specific day may be entered, as an integer from 0 to 99, forcing the model to use the maximum pipeline values through that day. Field: 2 character spaces; **T** (True) or **F** (False) or an integer. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Purchase peak pipelines (T/F) or max thru a given day:**]

BUYTOTAL — Definition: The buy total is the number of all components expected to be repaired at the base during the model run timeframe. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Minimum** and **Maximum:** 7 numeric spaces (2 decimal places) each; **Average**: 10 numeric spaces (2 decimal places); **Sum**: 10 numeric spaces. Output location: Statistics Window

Calculate button — Definition: Selecting the **Calculate** button will determine the new field value for each record that meets the selection criteria. This button will cause ISAAC to perform the operation specified in the **Change Type** drop-down list box using the value entered in the **Change Amount** text box and applying it to the selected field record values. Input location: Define Changes page of Sensitivity Change page frame.

Cancel Kit button — Definition: Selecting this button will cancel the kit and return you to the initial ISAAC screen. Selecting this button will delete the kit that you have created with the **Global Changes (Sensitivity) to Kit** option form the **Kit** pull-down menu. Input location: Sensitivity Change Screen

Cann1 — Definition: Cannibalization policy in force through the first analysis day. F' indicates full or all LRUs are cannibalized, 'P' indicates partial or LRUs with their cannibalization field set to "Y" are cannibalized, and 'N' indicates no cannibalization of LRUs. Selecting a value of 'P' forces the model only to cannibalize items that are easy to cannibalize. Whether an item is easy or hard depends on how the user sets the item flag.. A spares mix built using any cannibalization optimization will be radically different from an equal cost mix optimized for non-cannibalization. Field: Drop-down list box consists of **Full**, **None** or **Partial**; 1 character space. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **Cannibalization (Thru 1st day)**]

Cann2 — Definition: Cannibalization policy in force from the end of the first day to be analyzed through the second analysis day. F' indicates full or all LRUs are cannibalized, 'P' indicates partial or LRUs with their cannibalization field set to "Y" are cannibalized, and 'N' indicates no cannibalization of LRUs. Selecting a value of 'P' forces the model only to cannibalize items that are easy to cannibalize. Whether an item is easy or hard depends on how the user sets the item flag.. A spares mix built using any cannibalization optimization will be radically different from an equal cost mix optimized for non-cannibalization. Field: Drop-down list box consists of **Full**, **None** or **Partial**; 1 character space. Input location: Run Log

Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **Cannibalization (Thru 2nd day)**]

Cannflag — Definition: Specifies whether an LRU can be cannibalized or not. 'N' indicates that the item can not be easily cannibalized, while 'Y' indicates that the item can be easily cannibalized. *This field is only used when "Partial" is selected for the 'Cannibalization' field on the Model Parameters Screen.* With the '*Cannibalization' field* set to 'partial' the model will only cannibalize items that are easy to cannibalize. Field: Drop-down list box consists of **Y** (yes) or **N** (no). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Cannibalization Flag**]

Cannflag_1 — Definition: Specifies whether an LRU can be cannibalized or not. 'N' indicates that the item can not be easily cannibalized, while 'Y' indicates that the item can be easily cannibalized. SRUs will always have a 'Y' value in their Cannflag_1 field. When the model is run with the **Cannibalization:** field (for the first day to be analyzed) set to **Full,** all LRUs will have a 'Y' in this field. On the other hand, when the model is run with the **Cannibalization:** field (for the first analysis day) set to **None,** all LRUs will have a 'N' in this field. Field: 1 character space. Output location: Critical Item Window

Cannflag_2 — Definition: Specifies whether an LRU can be cannibalized or not. 'N' indicates that the item can not be easily cannibalized, while 'Y' indicates that the item can be easily cannibalized. SRUs will always have a 'Y' value in their Cannflag_1 field. When the model is run with the **Cannibalization:** field (for the second analysis day) set to **Full,** all LRUs will have a 'Y' in this field. On the other hand, when the model is run with the **Cannibalization:** field (for the second day to be analyzed) set to **None**, all LRUs will have a 'N' in this field. Field: 1 character space. Output location: Critical Item Window

Cannibalization (Thru 1st day): — Definition: Cannibalization policy in force through the 1st day to be analyzed. **Full** indicates that all LRUs may be cannibalized. **Partial** instructs the model to use the individual cannibalization field in each LRU record ('Y' LRUs are cannibalized, while 'N' indicates LRUs are not). **None** indicates that no LRU may be cannibalized. *Note: the model assumes that SRUs are always cannibalized in the repair shops.* Selecting a value of 'P' forces the model to only cannibalize items that are easy to cannibalize. Whether an item is easy or hard depends on how the user sets the item flag. A spares mix built using any cannibalization optimization will be radically different from one of an equal cost mix optimized for non-cannibalization. Field: Drop-down list box options of **Full**, **None**, or **Partial**. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Cann1] The model allows cannibalization of the 'Y' coded items and reports these actions as "easy" cannibalizations.

Cannibalization (Thru 2nd day): — Definition: Cannibalization policy in force from the end of the first day to be analyzed through the 2nd day to be analyzed. **Full** indicates that all LRUs may be cannibalized. **Partial** instructs the model to use the individual cannibalization field in each LRU record ('Y' LRUs are cannibalized, while 'N' indicates LRUs are not). **None** indicates that no LRU may be cannibalized. *Note: the model assumes that SRUs are always cannibalized in the repair shops.* Selecting a value of 'P' forces the model to only cannibalize items that are easy to cannibalize. Whether an item is easy or hard depends on how the user sets the item flag. A spares mix built using any cannibalization optimization will be radically different from an equal cost mix optimized for non-cannibalization. Field: Drop-down list box options of **Full**, **None**, or **Partial**. Input locations: Kit Parameters Screen; Model Parameters Screen [cross reference with Cann2]

Cannibalization Flag: — Definition: Specifies whether an LRU can be cannibalized or not. 'N' indicates that the item cannot be easily cannibalized, while; 'Y' indicates that the item can be easily cannibalized. *This field is only used when "Partial" is selected for the 'Cannibalization' field of the PARAMS file.* With the *'Cannibalization' field* set to 'partial' the model will only cannibalize items that are easy to cannibalize. Field: Drop-down list box consists of **Y** (yes) or **N** (no). Input location: Kit Component Data Screen [cross reference with Cannflag]

Cannind — Definition: Specifies whether an LRU can be cannibalized or not. 'N' indicates that the item can not be easily cannibalized, while; 'Y' indicates that the item can be easily cannibalized. Field: **Y** (Yes) or **N** (No), 1 character space. Output location: Pipeline Data Window

CDF — Definition: Cumulative Distribution Function Field: 8 numeric spaces (2 decimal places). Input location: Component Data Window. This field can not be edited within ISAAC.

Change Amount — Definition: This is the amount of change you want to apply to the field values. It is used in conjunction with the **Change Type** drop-down list box value. It is important that you enter a value in the **Change Amount** text box, even if the default value displayed is the value you want to apply. The screen Calculate and Reset buttons will not be activated until you have entered a value in the **Change Amount** field. Field: text box. Input location: Define Changes page of Sensitivity Change page frame.

Change Description: — Definition: Consists of the **Field** drop-down list box, **Change Type** drop-down list box, and the **Change Amount** text box. Input location: Define Changes page of Sensitivity Change page frame.

Change Type — Definition: This is the type of change you want to apply to the field values. The type of change available for any field is dependent upon the field type (i.e., character fields can be changed using a constant value and numeric fields can be changed using simple mathematical operations). It is used in conjunction with the **Change Amount** text box value. Field: drop-down list box consisting of **Constant** for character fields and **Constant**, **Add**, **Subtract**, and **Multiply** for numeric fields. Input location: Define Changes page of Sensitivity Change page frame.

Class — Definition: The classification of the component in terms of its reparability. Use 'R' for reparable items, 'A' for class A consumables, 'B' & 'C' for class B & C consumables respectively. Field: 1 character space (e.g., **R**, **A**, **B**, or **C**). Input location: Component Data Window. This field can not be edited within ISAAC.

Close button — Definition: Selecting this button closes this screen, while saving the information entered on the screen, and returns you to the previous screen. Input locations: Kit Parameters Screen; Kit Flying Hour Scenario Screen; Kit Pipeline, Resupply & Options Screen

– Selecting **Close** on the Model Parameters Screen or the Kit Parameters Screen will return you to the initial ISAAC screen. Input locations: Kit Parameters Screen, Model Parameters Screen

– Selecting **Close** on either the Model Flying Hour Scenario Screen or the Model Pipeline, Resupply & Options Screen will return you to the Model Parameters Screen. Input locations: Model Flying Hour Scenario Screen, Model Pipeline, Resupply & Options Screen

– Selecting **Close** on either the Kit Flying Hour Scenario Screen or the Kit Pipeline, Resupply & Options Screen will return you to the Kit Parameters Screen. Input locations: Kit Flying Hour Scenario Screen, kit Pipeline, Resupply & Options Screen

– Selecting **Close** on the Kit Component Data Screen will return you to the Kit Parameters Screen. When one or more changes are made and saved to the component data on the Kit Component Data Screen and you select **Close**, ISAAC will start the indenture checking process. Input location: Kit Component Data Screen

Comasset — Definition: Common assets Field: 7 numeric spaces. Input location: Component Data Window. This field can not be edited within ISAAC.

Comment — Definition: User entered descriptive text documenting information pertaining to the run. Field: text box, character. Input location: Model Parameters Screen [cross reference with Comment]

Comment — Definition: a way for the user to link descriptive information to a run after the run is made. Field: character (Memo Field). Input location: Run Log Window. Location to edit information: Model Parameters Screen. [cross reference with **Comment**]

Compipe — Definition: Common pipeline Field: 8 numeric spaces (4 decimal places). Input location: Component Data Window. This field can not be edited within ISAAC.

Component Data button — Definition: Selecting this button opens the Kit Component Data Screen. Input location: Kit Parameters Screen

Component Data Screen— Consists of all of the component level fields that define a kit. The Component Data Screen is the place where Basic field information (component data) and Conditional (Wartime / Non-wartime) Data can be edited. Basic field information consists of fourteen data fields that do not vary on the basis of wartime or non-war conditions. These field values are also unaffected by operational parameters such as flying hours. Conditional (Wartime / Non-wartime) Data fields vary on the basis of wartime or non-wartime conditions. These field values are affected by operational fields such as flying hours and / or maintenance fields such as failure rate. Each field has separate wartime and non-wartime values.

Condasset — Definition: The condemnation assets are those items expected to be condemned during the coverage period minus the items expected to be condemned during the items procurement lead time (PLTT). The Condasset value will be 0 for each item unless the model was run with the Use Starting Assets? field toggled to Use Assets: InitAsset + FreeAsset. When Condassets are not equal to 0 they are expressed as negative whole numbers. Field: 6 numeric spaces. Output Location: Shopping List Data Window

Condemnation Fraction — Definition: Condemnation fraction for this component. This is the fraction of base demand that cannot be repaired and must be condemned. Field: Text box with 8 numeric spaces (5 decimal places). Input location: Kit Component Data Screen [cross reference with Conpctp and Conpctw]

Condpipe — Definition: Condpipe is the condemnation pipeline mean. It is the expected number of components to be condemned during the item's procurement lead time total (PLTT). This is based on demand generated under the non-wartime flying hour program. Field: 8 numeric spaces (4 decimal places). Output location: Shopping List Data Window.

Conf_eval — Definition: The calculated confidence level of meeting the NMCS target, by day. Field: 8 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Confidenc1 — Definition: The achieved probability of meeting the first analysis day NMCS target expressed as a percent. There is a separate Confidenc1 value for each of the Budcodes and a Confidenc1 value that applies to all of the components that is called the Sum. The Sum value matches the Achieved Confidence of NMCS Target for the first analysis day (Day 1 Value). Field: 6 numeric spaces (2 decimal places). Output location: Performance Report Window [cross reference with Achieved Confidence of NMCS Target]

Confidenc2 — Definition: The achieved probability of meeting the second analysis day NMCS target expressed as a percent. There is a separate Confidenc2 value for each of the Budcodes and a Confidenc2 value that applies to all of the components that is called the Sum. The Sum value matches the Achieved Confidence of NMCS Target for the second analysis day (Day 2 Value). Field: 6 numeric spaces (2 decimal places). Output location: Performance Report Window [cross reference with Achieved Confidence of NMCS Target]

Confidnc1 — Definition: This is the probability of meeting the first analysis day NMCS target expressed as a percent. The model optimizes the probability that the number NMCS is not greater that the target. This is an optional input and is mutually exclusive of the **1st Budget** constraint (i.e., you may enter a confidence % or a budget target but not both). This is the value entered by the user in the "1st Confidence" field (if applicable). Field: 3 numeric spaces. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **1st Confidence**]

Confidnc2 — Definition: This is the probability of meeting the second analysis day NMCS target expressed as a percent. The model optimizes the probability that the number NMCS is not greater that the target. This is an optional input and is mutually exclusive of the **1st Budget** constraint (i.e., you may enter a confidence % or a budget target but not both). This is the value entered by the user in the "2nd Confidence" field (if applicable). Field: 3 numeric spaces. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **2nd Confidence**]

Conpctp — Definition: Non-wartime condemnation fraction for this component. This is the fraction of base demand that can not be repaired and must be condemned. Field: 8 numeric spaces (5 decimal places). Input location: Component Data window; Output location: View Input-Output Window. Location to edit information: Kit Component Data Screen. [cross reference with **Condemnation Fraction**]

CONPCTP — Definition: This is the non-wartime condemnation fraction for all components. This is the fraction of base demand that can not be repaired and must be condemned. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Average**, **Minimum**, and **Maximum:** 8 numeric spaces (5 decimal places) each; **Sum**: 10 numeric spaces (5 decimal places). Output location: Statistics Window

Conpctw — Definition: Wartime condemnation fraction for this component. This is the fraction of base demand that can not be repaired and must be condemned. Field: 8 numeric spaces (5 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Condemnation Fraction**]

Conpipe_1 — Definition: Conpipe_1 is the condemnation pipeline mean. It is the expected number of components to be condemned during the item's procurement lead time total (PLTT). Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

Conpipe_2 — Definition: Conpipe_2 is the condemnation pipeline mean. It is the expected number of components to be condemned during the item's procurement lead time total (PLTT). [XYX is this time period PLTT or since Day 1 of the war] Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

CONPIPE — Definition: Conpipe is the condemnation pipeline mean. It is the expected number of components to be condemned during each item's procurement lead time total (PLTT). Consists of four distinct parameters: average, minimum, maximum and sum. Fields —**Average**, **Minimum**, and **Maximum:** 7 numeric spaces (2 decimal places) each; **Sum**: 10 numeric spaces (2 decimal places). Output location: Statistics Window

Conpipe — Definition: Conpipe is the condemnation pipeline mean. It is the expected number of components to be condemned during the item's procurement lead time total (PLTT). A separate Conpipe value is calculated for each analysis day. The condemnation percentage for each component during non-war (Conpctp) and wartime (Conpctw) conditions are inputs to the model. Field: 7 numeric (2 decimal places). Output location: Pipeline Data Window.

Copy button — Definition: Selecting this button copies the highlighted record information to create a new component data record. Input location: Kit Component Data Screen

Cost Delta — Definition: The difference in item total buy cost between Model Runs 1 & 2. This is the absolute value of the difference between (BuyTot1 x BuyCost1) and (BuyTot2 x BuyCost2). Field: 8 numeric spaces. Output location: Shop Comparison Window.

Cost: — Definition: Unit cost of the component in US Dollars. Field: Text box with 11 numeric spaces (2 decimal places). Input location: Kit Component Data Screen [cross reference with Cost]

Cost_adjus — Definition: This is the model final Buy Cost (Cost_sum field) after the model has adjusted the original buy quantity to try and match the budget. There is one value for each order year of the buy. This field will be 0 except when the user changes the budget value in the "Spares Cost Summary a PLTT before Spares Delivery" Screen. In that case this field will contain the estimated spares purchase cost on the basis of the user adjusted annual budget. Field: 9 numeric spaces (2 decimal places). Output location: Yearly Cost Window

Cost_sum — Definition: This is the initial estimated annual spares purchase cost for each year from the aircraft delivery year back to the 1st year during which spares will be ordered. Field: 16 numeric spaces (2 decimal places). Output location: Yearly Cost Window

Cost — Definition: Unit cost of the component in US Dollars.

- Field: 11 numeric spaces (2 decimal places). Input location: Component Data window; Output locations: Critical Item and View Input-Output Windows. Location to edit information: Kit Component Data Screen. [cross reference with **Cost**]

– Field: 12 numeric spaces (2 decimal places). Output locations: Pipeline Data and Shopping List Data Windows. Location to edit information: Kit Component Data Screen. [cross reference with **Cost**]

Coverage Period — see Spares **Coverage Period:**.

Currently Selected Run - Buy_Total Field from Shopping List — see Select Spares Mix From...

Currently Selected Run - Recommended Buy (RMSSBUY) Field from Kit — see Select Spares Mix From...

Curve-val1 — Definition: The budget constraint for the first analysis day (in dollars). Field: 11 numeric spaces (2 decimal places). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **1st Budget**]

Curve-val2 — Definition: The budget constraint for the second analysis day (in dollars). Field: 11 numeric spaces (2 decimal places). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **2nd Budget**]

Date

– Definition: The date of the model run. Field: numeric date (e.g., April 15, 1998 expressed as 04/15/98). Input location: Model Parameters Screen [cross reference with Rdate]

– Definition: The date that the kit was created. Field: numeric date (e.g., April 15, 1998 expressed as 04/15/98). Input location: Kit Parameters Screen

Day 01 – 60 — Definition: The flying hour program per day for all weapon systems of interest. This field value will be divided by the Number of Bases and Number (of Weapon Systems) fields to determine the number of flying hours per day per aircraft. The scenario encompasses both the non-wartime and wartime flying hour programs. Only one value may be entered for the non-wartime flying hour program. Multiple values may be entered for the wartime flying hour program — however, only one value may be entered for each day. Field: Text box with 7 numeric spaces (2 decimal places). Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with **Day**]

Day Base Repair Begins — Definition: User-specified run time parameters that affect when base repair begins during wartime for RR LRUs, RRR LRUs, and SRUs. The user specifies whether an LRU is RR and RRR in the item's maintenance concept (MAINTCON) field. (Currently, the RR and RRR classification have no other meaning than to separate LRUs into 2 groups.) Entering an integer value that is greater than the last wartime analysis day ensures no repair of that category (e.g., for a 60 day war, '3' indicates repair starts on day 3; '61' indicates no repair is performed within the war, because repair starts on day 61). Field: Text box with 2 character spaces. Input location: Kit Pipeline, Resupply & Options Screen; Model Pipeline, Resupply & Options Screen [cross reference with Br_rrlru, Br_rrrlru, Br_sru for RR LRUs, RRR LRUs, and SRUs, respectively

Day Depot Repair Begins — Definition: User-specified run time parameters that affect when depot repair begins during wartime for RR LRUs, RRR LRUs, and SRUs. The user specifies whether an LRU is RR and RRR in the item's maintenance concept (MAINTCON) field. (Currently, the RR and RRR classification have no other meaning than to separate LRUs into 2 groups.) Entering an integer value that is greater than the last wartime analysis day ensures no repair of that category (e.g., for a 60 day war, '3' indicates repair starts on day 3; '61' indicates no repair is performed within the war, because repair starts on day 61). Field: Text box with 2 character spaces. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Br_rrlru, Br_rrrlru, Br_sru for RR LRUs, RRR LRUs, and SRUs, respectively

Day Order and Ship Begins: — Definition: The day when forward transportation from the depot starts. Field: Text box with 2 character spaces. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Os_start]

Day_1 — Definition: Day_1 is the first analysis day. It is taken from the **1st Analysis Day** field of the Parameters Screen. Field: 2 character spaces. Output location: Critical Item Window. [cross reference with **1st Analysis Day**]

Day_2 — Definition: Day_2 is the second analysis day. It is taken from the **2nd Analysis Day** field of the Parameters Screen. Field: 2 character spaces. Output location: Critical Item Window. [cross reference with **2nd Analysis Day**]

Day —

– Definition: The day that the pipeline component data pertains to. The Pipeline Data window will contain one record for each component for the number of days to be analyzed. If there are two days to be analyzed (.e.g., 1st Analysis Day = 0 and 2nd Analysis Day = 24) then each component will have a record for Day 0 and a record for Day 24. Field: 2 character spaces. Output location: Pipeline Data Window

– Definition: The analysis day with 0 representing peacetime and all other numbers representing the respective day of the war. Field: 8 numeric spaces. Output location: Multi-Day Evaluation Window

Day1+Day2 Cost — Definition: This is the total spares purchased cost (sum of both analysis days) required to achieve the corresponding ENMCS curve values. It is the sum of the Day 2 Cost value displayed plus the Day 1 Cost value (which is not shown). Field: 14 character spaces. Output location: Curve Window

Day1 — Definition: The first analysis day. A “0” for the model’s analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A “24” translates into having the model procure spares for a 24-day war scenario that starts at the end of the coverage period and ends 24 days later (alternatively, any other day of the wartime scenario could be entered). Field: 2 character spaces. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with**1st Analysis Day**]

Day2 Cost — Definition: This is the spares purchased cost required to achieve the corresponding ENMCS curve values for the last analysis day (for a 1 analysis day run this is Day 1's cost). Field: 14 character spaces. Output location: Curve Window

Day2 — Definition: The second analysis day. A “0” for the model’s analysis day translates into having the model procure spares for steady-state (non-war) conditions over the entire coverage period. A “24” translates into having the model procure spares to cover failures from the first analysis day up through the 24th day of the wartime scenario (that starts at the end of the coverage period). For the second analysis day of analysis, the model considers those spares bought on the first day as being in the inventory. It then purchases additional spares to meet the specified second-day target or second-day incremental budget constraint (if applicable). Thus, the spares that the model selects for the second analysis day make up the difference between the previous days' buy and those required to meet the second target. A blank (shown as an "*") indicates no second analysis day. Field: 2 character days. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **2nd Analysis Day**]

Decelerate Hrs.. — Definition: This field must be activated in order to use the decelerated flying hour capability of ISAAC to translate non-wartime demand into wartime demand. When this field is clicked, the**Factor** text box field is activated. Field: Check box. Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen. [cross reference with Usedecel]

Decelerate War Flying Hours?— Consists of the **Decelerate Hrs..** and **Factor** fields. Input locations: Kit Flying Hour Scenario Screen, Model Flying Hour Scenario Screen

Decelerate — Definition: This is where the value entered in**Factor** is recorded. The **Decelerate Hrs..** check box must be clicked before this field can be accessed. Field: 6 numeric spaces (3 decimal places). Input location: Run Log Window. [Cross reference with**Factor**]

Decon — Definition: This is the deceleration conversion or the ratio of the decelerated to the actual sortie duration. It is expressed mathematically as (Dhr_sortie / Hr_sortie).

– Field: 6 numeric spaces (3 decimal places). Input location: Run Log Window.

– Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Define Changes Page — The Define Changes page is the vehicle for determining which item records to change and what the nature and extent of the change will be. You can make changes to all records or to a user-specified subset of records. Input location: Sensitivity Change page frame.

Definition: Expected back orders (EBOs) for the item during the period of interest. For an analysis day of "0" , it is the EBOs at the end of the coverage period and for wartime analysis days, it is the EBOs for that day. Field: 7 numeric spaces (2 decimal places). Output location: Pipeline Data Window

– Definition: The total expected back orders (EBOs) for all items given the spares cost displayed in the Day 1 + Day 2 Cost column. Field: 7 character spaces. Output location: Curve Window

Delete Button —

– Definition: Selecting this button will delete all records associated with this particular **Run Description:**. This includes all output information from the run and the global input information (input via the Model Parameter, Model Flying Hour Scenario, and Model Pipeline, Resupply, & Options Screens and maintained in the **Run Log**). If you want to delete many runs, we suggest using delete in conjunction with the buttons to the right and left of the **Prev Runs** button. Note, the item-level input information cannot be deleted by using this delete button but only through the Kit menus. Input locations: Kit Parameters Screen, Model Parameters Screen

– Definition: Selecting this button will delete all records associated with this particular **Kit**. This includes all global input information (input via the Kit Parameter, Kit Flying Hour Scenario, and Kit Pipeline, Resupply, & Options Screens) and the associated component level information from the Kit . Input location: Kit Parameters Screen

– Definition: Selecting this button will delete one highlighted record at a time. Input location: Kit Component Data Screen

Delta_adju — Definition: Delta adjusted is the dollar amount of the spares delayed for a specific year. Field: 9 numeric spaces. Output location: Yearly Cost Window

Demand_1 — Definition: Expected demand for this component on the first analysis day. The demand is the product of the expected number of failures for this item per flying hour (Toimdrp or Toimdrw as appropriate) multiplied by the daily flying hours for the fleet. Field: 8 numeric (5 decimal places). Output location: Critical Item Window.

Demand_2 — Definition: Expected demand for this component on the second analysis day. The demand is the product of the expected number of failures for this item per flying hour (Toimdrp or Toimdrw as appropriate) multiplied by the daily flying hours for the fleet. Field: 8 numeric (5 decimal places). Output location: Critical Item Window.

Depot Repair Time — Definition: Depot repair time in days for this component. This is the number of days of the complete depot repair cycle from NRTS of the item from the base to the depot through repair of the item at the depot. This includes the retrograde ship time from the base to the depot; but, it does not include the shipping time from repair until the item is received at the base (order and ship time). Field: Text box with 6 numeric spaces. Input location: Kit Component Data Screen [cross reference with Idrtp and Idrtw]

Deppipe_1 — Definition: Deppipe_1 is a depot pipeline mean. It is the number of components expected to be in repair at the depot on the first analysis day. Field: 7 numeric (2 decimal places). Output location: Critical Item Window.

Deppipe_2 — Definition: Deppipe_2 is a depot pipeline mean. It is the number of components expected to be in repair at the depot on the second analysis day. Field: 7 numeric (2 decimal places). Output location: Critical Item Window.

DEPPIPE — Definition: Depot pipeline is the number of all components expected to be repaired at the depot during the model run timeframe. Consists of four distinct parameters: average, minimum, maximum and sum. Fields – **Average**, **Minimum**, and **Maximum:** 7 numeric spaces (2 decimal places) each; **Sum: 10 numeric spaces (2 decimal places).** Output location: Statistics Window

Deppipe — Definition: Deppipe is the depot pipeline mean. It is the number of components expected to be in repair at the depot for a specific analysis day. The depot repair time for each component during non-war (Idrtp) and wartime (Idrtw) conditions are key inputs to the model. Field: 7 numeric spaces (2 decimal places). Output location: Pipeline Data Window

Describe — Definition: This is the descriptive information entered in the "Run Description" field by the user. Field: 40 character spaces. Input location: Run Log Window. Location to edit information: Model Parameters Screen. [cross reference with **Run Description**]

Description: — Definition: This is entered by the user during the kit creating process. The **Description** should be meaningful to you so as to aid in future identification of this kit. Location to input information: Define Kit ID dialog box. Input location: Kit Parameters Screen

Dfh_flown — Definition: is the decelerated flying hours flown (see Deceleration section). Field: 8 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Dfh_plan — Definition: is the decelerated flying hour plan (see Deceleration section). Field: 8 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Dhr_sortie — Definition: is the decelerated number of hours per sortie. If the non-war flying hour duration is not exactly one or the wartime duration is not a whole number, the model uses a proportional formula to estimate the deceleration duration. Field: 11 numeric spaces (2 decimal places). Output location: Multi-Day Evaluation Window

Dr_rrlru — Definition: the day that depot repair starts for RR repaired LRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Depot Repair Begins**]

Dr_rrrlru — Definition: the day that depot repair starts for RRR repaired LRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Depot Repair Begins**]

Dr_sru — Definition: the day that depot repair starts for SRUs. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Day Depot Repair Begins**]

Drt — Depot Repair Time (see Idrtp & Idrtw)

Earlpct — Definition: The percentage of Not Reparable at This Station which is done early (i.e., the percentage of failed items that are NRTSed directly to depot without going through the base repair cycle). Percentage must be 100% (item sent directly to depot with no delay at base) or 0% (item waits a base repair time before being sent to the depot for repair. Field: 5 numeric spaces (2 decimal places) (e.g., **1.00** or **0.00**). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Early NRTS**]

Early NRTS: — Definition: The percentage of Not Reparable at This Station which is done early (i.e., the percentage of failed items that are NRTSed directly to depot without going through the base repair cycle). A percentage of 100% (entered as **1**) means the item is sent directly to depot with no delay at base (the standard assumption used for most items). A percentage of 0% (entered as **0**) means an item waits a base repair time (isolating the problem) before being sent to the depot for repair (an item where this happens is an exception). Field type: Drop-down box consisting of 1 = Standard or 0 = Exception. Input location: Kit Component Data Screen

Ebo_qpa_1 -- This field equals the Ebos field divided by the Iqpa field for all cannibalization (CannFlag = 'Y') items for the first analysis day. For non-cannibalization items it is equal to zero. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

Ebo_qpa_2 — Similar to Ebo_qpa_1 but applies to the second analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

Ebos_1 — Definition: The number of expected back orders for the item on the first analysis day. For an analysis day of "0" , it is the EBOs at the end of the coverage period and for wartime analysis days, it is the EBOs for that day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Ebos_2 — Definition: The number of expected back orders for the item on the second analysis day. For an analysis of Day 0 , it is the EBOs at the end of the coverage period and for a wartime analysis day, it is the EBOs for that day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Ebos_eval — Definition: The calculated expected back orders, by day. This is the total number of EBOs across all items under consideration. Field: 8 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Ebos —

Ebos1 — Definition: The number of expected back orders (for all items that were considered) on the first analysis day. The total quantity of spares purchased to support both days to be analyzed are considered in the calculation. For an analysis of Day 0, it is the EBOs at the end of the coverage period and for a wartime analysis day, it is the EBOs for that day. Consists of one distinct field value for each budcode and a SUM across all budcodes. This SUM matches the Day 1 Values Expected Back Orders. Field: 7 numeric spaces (2 decimal places). Output location: Performance Report Window. [cross reference with Expected Back Orders]

Ebos2 — Definition: The number of expected back orders (for all items that were considered) on the second analysis day. The total quantity of spares purchased to support both days to be analyzed are considered in the calculation. For an analysis of Day 0, it is the EBOs at the end of the coverage period and for a wartime analysis day, it is the EBOs for that day. Consists of one distinct field value for each budcode and a SUM across all budcodes. This SUM matches the Day 2 Values Expected Back Orders. Field: 7 numeric spaces (2 decimal places). Output location: Performance Report Window. [cross reference with Expected Back Orders]

Edit button — Definition: Selecting this button enables you to edit existing component records, one at a time. Input location: Kit Component Data Screen

Enmcs_eval — Definition: The model calculated expected not mission capable for supply. This is calculated separately for each day that has been evaluated in the multi-day evaluation run. Field: 8 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Enmcs — Definition: The total number of weapon systems (e.g. aircraft) that are expected to be not mission capable for supply on the second analysis day for a two day analysis or the first analysis day for a one day analysis, given the spares cost displayed in the Day 1 + Day 2 Cost column. Field: 10 character spaces. Output location: Curve Window

Enmcs1 — Definition: The number of weapon systems (e.g., aircraft) that are expected to be not mission capable, due to a shortage of spares, on the first analysis day. Consists of one distinct field value for each budcode and a SUM across all budcodes. This SUM matches the Day 1 Values Expected Back Orders. Field: 9 numeric (4 decimal places). Output location: Performance Report Window. [cross reference with Expected Not Mission Capable for Supply]

Enmcs2 — Definition: The number of weapon systems (e.g., aircraft) that are expected to be not mission capable, due to a shortage of spares, on the second analysis day. Consists of one distinct field value for each budcode and a SUM across all budcodes. This SUM matches the Day 2 Values Expected Back Orders. Field: 9 numeric (4 decimal places). Output location: Performance Report Window. [cross reference with Expected Not Mission Capable for Supply]

Enter a Range of Days and the Value to Store dialog box — Consists of the **First day:**, **Last Day:** and **Value:** fields. This dialog box gives you the capability to set wartime flying hours for a range of days simultaneously. The constraint is that the flying hour value will be constant over the range of days. Input locations: Kit Flying Hour Scenario Screen and Model Flying Hour Scenario Screen

Existing Assets — (see Initasset)

Expected Back Orders — Definition: Expected back orders (EBOs) for all items for the respective analysis day. For an analysis day of "0", it is the EBOs at the end of the coverage period and for wartime analysis days, it is the EBOs for that day. Field: 9 numeric spaces (2 decimal places). Output location: Performance Report Window. [cross reference with Ebos1 and Ebos2]

Expected Not Mission Capable for Supply — Definition: The number of weapon systems (e.g., aircraft) that are expected to be not mission capable, due to a shortage of spares, on the respective day(s) to be analyzed. Field: 9 numeric spaces (4 decimal places). Output location: Performance Report Window [cross reference with Enmcs1 and Enmcs2]

Exponential Repair: — Definition: Specifies whether wartime repair times are exponentially distributed ('T') or fixed ('F'). If 'F', all repairs are deterministic and take exactly the repair time set in the component data. If 'T', the repair time for a specific item varies according to an exponential distribution with the mean equal to the items repair time in the component data. Field: Text box with 1 character space. Kit Pipeline, Resupply & Options Window; Model Pipeline, Resupply & Options Window. [cross reference with Expresup]

Expresup — Definition: This is the value from the "Exponential Repair" field. T (true) will instruct the model to treat the repair time transition from non-wartime to wartime as an exponential process; while, F (false) will cause the model to treat it as a deterministic process. Field: 1 character space (e.g.,**T** or **F**). Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Exponential Repair:**]

Factor — Definition: This is the deceleration conversion or the ratio of the decelerated sortie duration to the actual sortie duration (see Deceleration section). The **Decelerate Hrs..** check box must be clicked before this field can be accessed. Field: Text box with 6 numeric spaces (3 decimal places). Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen. [cross reference with Decon]

Failure Based Demand — Definition: Demand per flying hour for this component. The demand is based on the expected number of failures for this item. An item is classified as failing if it can only be recovered through procurement or repair. The field has two elements a non-war and a wartime field value for each component. Field: Text box with 8 numeric spaces (5 decimal places). Input location: Kit Component Data Screen [cross reference with Toimdrp and Toimdrw]

Fap — Definition: Future application percentage (FAP): the percentage of aircraft that will be configured with this NSN. For common SRUs, this is the percentage of weapon systems that have this particular SRU on its respective LRU (Next Higher Assembly). For example, if a common SRU is on 50% of LRU A and LRU A is on 40 percent of the weapon systems then the user should enter a 0.2 FAP value (0.5x0.4) for that common SRU record. Field width: 6 numeric spaces (2 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Future Application:**]

Fh_flown — Definition: This is the number of flying hours that the fleet could sustain given the constraints of **Max. Sorties/Day:**, **Flying Hrs/Sortie:**, and the number of aircraft expected to be available to on the particular day of the multi-day evaluation. The value is calculated as the minimum of the planned flying hours and (Number of Aircraft - ENMCS) * (Hours / Sortie) * (Max.Sorties / day). Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Fh_plan — Definition: The planned flying hours for the fleet, by day. This is the number of flying hours from the Flying Hour Scenario Screen, for each day of the multi-day evaluation. Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window [cross reference with **Total Flying Hours:** (for non-wartime flying hours) and **Day 01 – 60** (for wartime flying hours)]

Field — Definition: This is a drop-down list box that contains a list of item record fields. There are two different lists of fields. The first list applies to the Change Description: line and consists of all the fields that you can make global changes to. The second list applies to the Filter Condition 1 and the Filter Condition 2 lines and consists of all the fields from the first list plus some identification fields that you are not allowed to make global changes to (e.g., NSN, NHA). Input location: Define Changes page of Sensitivity Change page frame.

Filter Condition 1 — Definition: Filter Condition 1 consists of the **Field** drop-down list box, **Filter Type** drop-down list box, and the **Filter Value** text box. [Filter Condition 1 is deselected until **Apply Change To:** is set to **Filter**] Input location: Define Changes page of Sensitivity Change page frame.

Filter Type — Definition: This is the logical filter you want to apply to the field values. Field: Drop-down list box consisting of **Equal To**, **Not Equal To**, **Less Than**, **More Than**, **Between**, and **Not Between** for numeric fields and **Equal To,** and **Not Equal to** for character fields. [Filter Type is deselected until **Apply Change To:** is set to **Filter**] Input location: Define Changes page of Sensitivity Change page frame.

Filter Value —Definition: The filter value that you want to apply. There is generally one filter value per filter condition; except, when the Filter Type is **Between** or **Not Between**. In the latter cases you must enter two separate filter values into two separate boxes. The lowest numerical value should be entered in the upper text box. [Filter Value is deselected until **Apply Change To:** is set to **Filter**] Input location: Define Changes page of Sensitivity Change page frame.

Filter Condition 2 — Definition: Filter Condition 2 consists of the **Field** drop-down list box, **Filter Type** drop-down list box, and the **Filter Value** text box. [Filter Condition 2 is deselected until **Apply Change To:** is set to **Filter**] Input location: Define Changes page of Sensitivity Change page frame.

Filter 1:— Definition: This read-only text box contains a mathematical expression that relates the three Filter Condition 1 variables: **Field**, **Filter Type** and **Filter Value.** The values that were selected on the Define Changes page are displayed in this text box. [Filter 1 is blank unless **Apply Change To:** is set to **Filter**.] Input location: Preview /Edit /Apply page of Sensitivity Change page frame.

Filter 2: — Definition: This read-only text box contains a mathematical expression that relates the three Filter Condition 2 variables: **Field**, **Filter Type** and **Filter Value.** The values that were selected on the Define Changes page are displayed in this text box. [Filter 2 is blank unless **Apply Change To:** is set to **Filter** and **Link** is set to **AND** or **OR**.] Input location: Preview /Edit /Apply page of Sensitivity Change page frame.

Filter1 — Definition: This column contains the field values pertaining to the Filter Condition 1 **Field** that was selected on the Define Changes page. [Filter1 is not displayed unless **Apply Change To:** is set to **Filter**.] Input location: Preview /Edit /Apply page of Sensitivity Change page frame.

Filter2 — Definition: This column contains the field values pertaining to the Filter Condition 2 **Field** that was selected on the Define Changes page. [Filter2 is not displayed unless **Apply Change To:** is set to **Filter** and **Link** is set to **AND** or

OR.] Input location: Preview /Edit /Apply page of Sensitivity Change page frame.

Find Kit button — Definition: Selecting the **Find Kit** button opens the Available Baseline Kits Window which will enable you to choose another baseline kit to work with. Input location: Kit Parameters Screen [Note: The two buttons on either side of the **Find Kit** button can be used to move around the library of previous kits without viewing them through the Available Baseline Kits Window.] The buttons are defined as follows:

- Selecting the **>** button will fill the Kit Parameters Screen with the global parameters from the previous kit that was immediately above the currently displayed kit in the Kit Library.
- Selecting the **>>** button will fill the Kit Parameters Screen with the global parameters from the kit at the top of the Kit Library.
- Selecting the **>** button will fill the Kit Parameters Screen with the global parameters from the previous kit that was immediately above the currently displayed kit in the Kit Library.
- Selecting the **<<** button will fill the Kit Parameters Screen with the global parameters from the kit at the bottom of the Kit Library.

First Day: — Definition: This is the text box where the user enters the first war day of a period of constant fleet daily flying hours. The text box will accept one whole number from 1 (the first day of the war) through 99 (Day 99 of the war). Field: Text box with 2 numeric spaces. Input location: Enter a Range of Days and the Value to Store dialog box (accessed through the Flying Hour Scenario Screen).

Flying Hrs/Sortie: — see Non-Wartime **Flying Hrs/Sortie:** and Wartime **Flying Hrs/Sortie:**

Force Buy Based on Pipeline % Below: — This sub-section enables users to force the model to procure spares quantities based upon an item's pipeline. It consists of 5 text boxes that impact purchase relative to item pipelines (**LRU % on first day:**, **LRU % on second day:**, **SRU % on first day:**, **SRU % on second day:**, and **Purchase peak pipelines (T/F) or max thru a given day**). Input location: Stock Options Section of the Pipeline, Resupply, & Options Screen.

Forced Buys - Cost Included in Total Spares Cost — see Use Spares Mix as...

Freeasset — Definition: Free assets is an ISAAC specific term developed to incorporate common components into the spares calculation. Common components are components common to both the initial provisioning aircraft and other IAF aircraft. Treatment of common components must apply any surplus stock already in the IAF inventory toward the requirement and must consider economies of scale. Since stock is already available for other aircraft, the new aircraft needs less inventory than if the part were not common. ISAAC uses a simple approximation to incorporate those benefits that are expressed in terms of "free assets": assets available to the item free of charge because of its commonality characteristics. Field: 7 numeric spaces. Output location: Shopping List Data Window

Future Application: — Definition: Future application percentage: the percentage of aircraft that will be configured with this NSN. For common SRUs, this is the percentage of weapon systems that have this particular SRU on its respective LRU (Next Higher Assembly). Field: Text box with 6 numeric spaces (2 decimal places). Input location: Kit Component Data Screen [cross reference with Fap]

Hr_sortie — Definition: This is the planned number of flying hours per sortie for the particular day of the multi-day evaluation. Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Iafid — Definition: IAF identification number (if applicable). Field: 9 character spaces. Input location: Component Data Window. Output location: View Input-output Window. This field can not be edited within ISAAC.

Ibrtp — Definition: Non-wartime base repair time (BRT) in days for this component. This is the number of days, in non-wartime, from item failure through the repair of the item to a serviceable status. Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Base Repair Time**]

IBRTP — Definition: Non wartime base repair time for all components. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Minimum** and **Maximum:** 6 numeric spaces each; **Average** and **Sum**: 10 numeric spaces (2 decimal places) each. Output location: Statistics Window

Ibrtw — Definition: Wartime base repair time (BRT) in days for this component. This is the number of days, in wartime, from item failure through the repair of the item to a serviceable status. Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Base Repair Time**]

Ibudcode — Definition: A user defined budget code from 1 through 99 which permits cost and availability subtotals to be generated by budget code. Ibudcode is envisioned to be used to identify all items that are part of a given subsystem (e.g., fire control , landing gear, etc.) so as to be able to predict subsystem parameters. Field: 2 numeric spaces. Input location: Component Data Window; Output location: View Input-output Window. Location to edit information: Kit Component Data Screen. [cross reference with **Budget Code:**]

Icurve — Definition: Iterate curve. NOT USED BY ISAAC. Field: 1 character space. Input location: Run Log Window.

Idealordyr — Definition: the year the item is ideally ordered. That year is the aircraft delivery year minus the PLTT (in years and rounded up). Field: 8 numeric spaces (2 decimal places). Output location: Shopping List Data and Yearly Cost Windows

IDRTP — Definition: Non-wartime depot repair time for all components. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Minimum** and **Maximum:** 6 numeric spaces each; **Average** and **Sum**: 10 numeric spaces (2 decimal places) each. Output location: Statistics Window

Idrtp — Definition: Non-wartime depot repair time (DRT) in days for this component. This is the number of days, in non-wartime, of the complete depot repair cycle from NRTS of the item from the base to the depot through repair of the item at the depot. This includes the retrograde ship time from the base to the depot; but, it does not include the shipping time from repair till the item is received at the base (order and ship time). Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Depot Repair Time**]

Idrtw — Definition: Wartime depot repair time (DRT) in days for this component. This is the number of days, in wartime, of the complete depot repair cycle from NRTS of the item from the base to the depot through repair of the item at the depot. This includes the retrograde ship time from the base to the depot; but, it does not include the shipping time from repair till the item is received at the base (order and ship time). Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Depot Repair Time**]

Include Starting Assets ? — Definition: This parameter specifies whether starting stock is considered in the computations Field: Drop-down list box consists of **Use Assets - Item's Itasse** or **Do NOT Use Starting Assets**. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen.

Initasset — Definition: The starting asset position for the NSN before any buys are made by ISAAC. A value in this field will not impact the budget estimates. The value in this field will only impact the availability estimates if the **Include Starting Assets?** field of the Parameters Screen is set to **Use Assets - Item's Itasse**. Field: 7 numeric spaces. Input location: Component Data Window. Output locations: Critical Item, Shopping List Data and View Input-Output Windows. Location to edit information: Kit Component Data Screen. [cross reference with **Initial Assets:**].

Initial Assets: —

– Definition: The starting asset position for the NSN before any buys are made by ISAAC. A value in this field will not impact the budget estimates. The value in this field will only impact the availability estimates if the **Include Starting Assets?** field of the Parameters Screen is set to **Use Assets - Item's Itasse**. Field: Text box with 7 numeric spaces. Input location: Kit Component Data Screen [cross reference with InitAsset]

– Definition: The dollar value of all item assets specified in the InitAsset field for each item. Field: numeric. Output location: Performance Report Window

Initial Assets - Spares are Free and NOT Included in Total Spares Cost — see Use Spares Mix as...

Initpltt — Definition: Procurement leadtime for the item in months. The Initpltt is the time from when an item is condemned to when a serviceable replacement for the item is procured and available at the base. The Initpltt can be thought of as the sum of the administrative leadtime required to order the item once the failure is discovered, the production leadtime and the time required to process and ship the item. Field: 6 numeric spaces. Output location: Shopping List Data Window

Iostp — Definition: Non-wartime order and ship time (OST) in days for this component. This is the number of days, in non-wartime, from when a request is made on the depot for an item until that item is received in base supply. This does not include depot shortage time when a serviceable item is not available at the depot. Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Order & Ship Time**]

IOSTP — Definition: Non-wartime order and ship time (OST) in days for all components. Fields — **Minimum** and **Maximum:** 6 numeric spaces each; **Average** and **Sum**: 10 numeric spaces (2 decimal places) each. Output location: Statistics Window

Iostw — Definition: Wartime order and ship time (OST) in days for this component. This is the number of days, in wartime, from when a request is made on the depot for an item until that item is received in base supply. This does not include depot shortage time when a serviceable item is not available at the depot Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Order & Ship Time**]

Iqpa — Definition: Quantity per aircraft of the component. This is the total quantity of this component on the aircraft (weapon system) assuming all NHA FAPs equal one. Field: 4 numeric spaces. Input location: Component Data Window; Output locations: Critical Item and View Input-Output Windows. This field can not be edited within ISAAC.

Itasse — Definition: If the **Include Starting Assets?** field of the Parameters Screen is set to **Use Assets - InitAsset + FreeAsset** it is the initial asset position plus any free assets minus any condemnations that are expected to occur during the coverage period. If the **Include Starting Assets?** field of the Parameters Screen is set to **Do NOT Use Starting Assets** then the ITASSE value will be zero. The value in this field will always impact the availability estimates. Field: 7 numeric spaces. Output location: Shopping List Data Window

Item Buy: — Definition: Percentage of the pipeline that is to be bought sacrosanct for this item. Applicable only when 'ITEM' is indicated as the PBUY percentage on the PARAMS file. (The PBUY percentage is entered on the Stock Options section of the Pipeline, Resupply, & Options Screen as LRU percentage on first day, SRU percentage on first day, etc.) Field: 5 numeric spaces (2 decimal places). Input location: Kit Component Data Screen [cross reference with Itembuy]

Item Name: — Definition: The noun nomenclature of the component part. Field: 19 character spaces. Input location: Kit Component Data Screen [cross reference with Itemname]

Itembuy — Definition: Percentage of the pipeline that is to be bought sacrosanct for this item. Applicable only when 'ITEM' is indicated in the respective **Force Buy based on Pipeline %** field on the Pipeline, Resupply, & Options Screen. (The **Force Buy based on Pipeline %** entered on the Stock Options section of the Pipeline, Resupply, & Options Screen as LRU percentage on 1st day, SRU percentage on 1st day, etc.) Field: 5 numeric spaces (2 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Item Buy:**]

Itemized Performance - By BUDCODE — Consists of the following output fields which are defined: Budcode, Buycost1, Confidenc1, Enmcs1, Ebos1, Totlrus1, Buycost2, Confidenc2, Enmcs2, Ebos2, and Totlrus2. Output location: Performance Report Window

Itemname — Definition: The noun nomenclature of the component part. Field: 19 character spaces. Input location: Component Data Window; Output location: Critical Item Window. Location to edit information: Kit Component Data Screen. [cross reference with **Item Name:**]

Itempipe — Definition: This is the sum of the base pipeline mean (Baspipe value) plus the depot pipeline mean (Deppipe value) plus the order and ship pipeline mean (Ospipe) and the condemnation pipeline mean (Conpipe) from the Pipeline Data database by component for NON-WARTIME ONLY. This field always yields the non-wartime pipeline even if the two days to be analyzed are wartime days and does not include any AWP. Field: 8 numeric spaces (4 decimal places). Output location: Shopping List Data Window

Kit # — Definition: The kit # is a unique number assigned to each baseline kit for tracking and identification purposes in the library of available kits. (Same as Kit ID Number field.) Field: 6 character spaces. Input location: Model Parameters Screen, Output location: Performance Report Window

Kit ID Number — Definition: The kit ID number is a unique number assigned by ISAAC to each baseline kit for tracking and identification purposes in the library of available kits. (Same as Kit # field.) Field: 6 numeric spaces. Input location: Kit Parameters Screen, Sensitivity Change page frame. [cross reference with As_kitid]

Kit Description — Definition: The **Kit Description** field displays the information entered in the **Description** field of the Define Kit ID dialog box. This information is entered by the user during the kit creating process. Field: character spaces. Input location: Kit Parameters Screen, Sensitivity Change page frame. Output location: Performance Report Window [cross reference with As_kitdesc]

Kit Name — Definition: The kit name is a name associated with a particular kit. This name is entered by the user during the kit creating process. Field: character spaces. Input location: Define Kit ID dialog box. Output location: Performance Report Window [cross reference with As_kitname]

Kit Parameters — Consists of all of the global parameter fields that define a kit. Most of these fields are the same fields as in the Model Parameters. Kit Name, Weapon System, and Description are unique to the kit.

Last Day: — Definition: This is a text box that will accept one whole number from 1 (the first day of the war) through 99 (Day 99 of the war). The number entered in the text box must be greater than the number entered in the **First Day:** field. Field: 2 numeric spaces. Input location: Enter a Range of Days and the Value to Store dialog box (accessed through the Flying Hour Scenario Screen).

Last_chg — Definition: This is the last day that the flying hours change. After this point the flying hours are treated as steady state by the model (using the flying hours of this last change day). The model enters this value based on the entries on the Flying Hour Scenario Screen. Field: 2 numeric spaces. Input location: Run Log Window; Location to change information: Flying Hour Scenario (Kit or Model).

Level — Definition: Indenture level of the component. The indenture structure describes the relationship between components. The aircraft is composed of LRUs (1st level of indenture), LRUs are composed of specific SRUs (2nd level of indenture), and those SRUs are composed of lower indenture SRUs (3rd level of indenture), and so on. Field: 7 numeric spaces. Input location: Component Data Window; Output locations: Critical Item, Pipeline Data, Shopping List and View Input-Output windows. This field can not be edited within ISAAC.

Link —This is a drop-down list box consisting of **And** and **Or**. This field is used to specify the link between Filter Condition 1 and Filter Condition 2. It is only required to be used if both Filter Condition 1 and Filter Condition 2 are being used. [**Link** is deselected until **Apply Change To:** is set to **Filter**] Input location: Define Changes page of Sensitivity Change page frame.

LRU % on first day: — Definition: The percentage of the LRU pipeline (either peak or on the first analysis day as specified by the **Purchase peak pipelines (T/F) or max thru a given day:** value**)** that the model buys sacrosanct to meet the requirement for the first analysis day. A value of 1.0 buys the whole pipeline, .5 buys half the pipeline, etc. An entry of 'ITEM' instructs the model to use the item-specific percentage as coded on the component data record. An entry of 'QPA' buys the pipeline for all items with a QPA greater than two. Field: numeric

(1 decimal place), **ITEM**, or **QPA**; Input location: Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Pbuyl1]

LRU % on second day: — Definition: The percentage of the LRU pipeline (either peak or on the second analysis day as specified by the **Purchase peak pipelines (T/F) or max thru a given day:** value**)** that the model buys sacrosanct to meet the requirement for the second analysis day. Field: numeric (1 decimal place), **ITEM**, or **QPA**; Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Pbuyl2]

Lru_rank1 — This field is used to rank the LRUs in order of their criticality based on their expected backorders on the first analysis day. The lower the value the more critical the LRU is. This field is derived from Lru_value1. The LRU with the highest Lru_value1 will be assigned the Lru_rank1 of "1." This is the most critical LRU. The next highest will be assigned a rank of '2" and so on. The ranking will be a continuous sequence and is only designed to provide an order of criticality not a measure of criticality. SRUs will have the same Lru_rank1 as their LRU. Field: 4 numeric spaces. Output location: Critical Item Window.

Lru_rank2 — Similar to Lru_rank1 but applies to the second analysis day. Field: 4 numeric spaces. Output location: Critical Item Window.

Lru_value1 — This is a nominal value assigned to each LRU based on that LRU's backorders on the first analysis day. The higher the value the more critical the LRU is. This value is designed to provide a relative measure of criticality. For the cannibalization case, Lru_value1 = EBOs / QPA (field labeled Ebo_qpa_1). For the no cannibalization case, Lru_value1 = EBOS_1 (field labeled Ebo_qpa_1 equals 0). SRUs will have the same Lru_value1 as their LRU. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

Lru_value2 — Similar to Lru_value1 but applies to the second analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window.

Lrunsn **—** Definition: For each item, this field specifies what LRU the item is a sub-component of. Field: 18 character spaces. Input location: Component Data Window; Output location: Critical Item Window. This field can not be edited within ISAAC.

MAINTCON — This affects when (if ever) wartime base and depot repair of failed LRUs and SRUs begins. The maintenance concept is used to group spares for the purpose of establishing when repair begins for each group in wartime. Specifically the MAINTCON can be used to determine when base and depot repair begin during wartime for RR (remove and replace) LRUs, RRR (remove, repair, and replace) LRUs, and SRUs. The standard use of those categories assumes the RRR items have repair start early in the war and RR have no repair until later in the war. However, the user may specify an LRU as RR or RRR based upon their own definition that separates LRUs into any 2 groups of items that each have their repair start on different days of the war. The Resupply section of the Stock, Resupply, and OtherOptions Screen enables the user to enter the day repair starts (using the starting day of the war as the reference point) at the base and at the depot. A separate repair start date can be specified for RR or RRR LRUs and SRUs at the depot and at the base. Field width: 3 character spaces (e.g., **RR** or **RRR** for LRUs and **RRR** for SRUs). Input location: Component Data Window; Output location: Pipeline Data Window. Location to edit information: Kit Component Data Screen. [cross reference with **MAINTCON:**]

MAINTCON: — Definition: This affects when (if ever) wartime base and depot repair of failed LRUs and SRUs begins. The maintenance concept is used to group spares for the purpose of establishing when repair begins for each group in wartime. Specifically the MAINTCON can be used to determine when base and depot repair begin during wartime for RR (remove and replace) LRUs, RRR (remove, repair, and replace) LRUs, and SRUs. The standard use of those categories assumes the RRR items have repair start early in the war and RR have no repair until later in the war. However, the user may specify an LRU as RR or RRR based upon their own definition that separates LRUs into any 2 groups of items that each have their repair start on different days of the war.The Resupply section of the Stock, Resupply, &Options Screen enables the user to enter the day repair starts (using the starting day of the war as the reference point) at the base and at the depot. A separate repair start date can be specified for RR or RRR LRUs and SRUs at the depot and at the base. Field type: Drop down box consisting of **RR No War Repair** or **RRR War Repair**. Input location: Kit Component Data Screen

Manager: — Definition: Item manager code. Field: 4 character spaces. Input location: Kit Component Data Screen [cross reference with Manager]

Manager — Definition: Item manager code. Field: 4 character spaces. Input location: Component Data Window; Output location: View Input-output Window. Location to edit information: Kit Component Data Screen. [cross reference with **Manager:**]

Max Sorties/Day: **—** see Wartime **Max Sorties/Day:**

Max_adjust — Definition: Max adjusted represents the maximum number of dollars worth of spares that can have their orders delayed without effecting the availability results. Field: 16 numeric spaces (2 decimal places). Output Location: Yearly Cost Window. Location to edit information: Spares Cost Summary a PLTT Before Delivery Screen.

Max_sortie — Definition: The maximum number of sorties an aircraft can fly per day (user input entered in the Flying Hour Scenario Screen). This is also referred to as the "turn rate." Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window. [cross reference with **Max Sorties/Day:** field]

Maxsortie — Definition: The maximum number of sorties an aircraft can fly per day. This is also referred to as the "turn rate." Field: 6 numeric spaces (3 decimal places). Input location: Run Log Window. Location to edit information: Flying Hour Scenario Screen. [cross reference with **Max Sorties/Day:** field]

Model ID — Definition: Unique number assigned to each model run for tracking and identification purposes in the library of previous runs. Field: numeric. Output location: Performance Report Window

Model Parameters — Consists of all of the global parameter fields that define a model run. Most of these fields are the same fields as in the Kit Parameters. Weapon System, Run Description and User Name are unique to the model run.

Modify button— Selecting **Modify** in the Model Parameters Screen informs ISAAC that you are going to modify one or more of the model parameters (displayed on the screen from a previous run) to produce a new run. Selecting **Modify** in the Kit Parameters Screen informs ISAAC that you are going to modify one or more of the kit parameters (displayed on the screen from the current kit). Input locations: Kit Parameters Screen, Model Parameters Screen

Mru — Definition: The minimum replacement quantity of an item that is normally replaced / installed upon failure or scheduled service. Field: 3 numeric spaces. Input location: Component Data Window; Currently, this field can not be edited or viewed from within ISAAC.

Multiday — Definition: This is one of the two fields (the other field is Rb_runtype) where the selection(s) from the Permit More Buys? section of the Evaluation Setup Screen is stored. There are three possible values which are listed below. Field: 1 numeric space. Input location: Run Log Window. Location to edit information: Evaluation Setup Screen. [cross reference with Permit More Buys?]

- 0 =
  - **Requirement Run**. This field will have a value of zero for all requirement model runs.
  - "**Yes — Use Mix as Starting Point (Add Spares if Needed to Reach Target)**". The Multiday field will have a value of zero whenever this analysis type has been selected. The model will add to the selected spares mix, if needed, to achieve the availability or cost targets specified on the Parameters Screen.
- 1 = "**No — Evaluate Selected Spares Mix Only**" + "**Standard 1 or 2 Day Run**". The Multiday will have a value of one whenever the model is run with these evaluation run options selected. This is the standard evaluation run setting.
- 2 = "**No — Evaluate Selected Spares Mix Only**" + "**Multi-day - All Days Inclusive (0,1,2,3,4...)**". The Multi-day will have a value of one whenever the model is run with these evaluation run options selected. This will result in a multi-day evaluation run of the model. The model will evaluate the availability that this kit yields on each consecutive day to be analyzed. The model will evaluate the kit from **Day 1** to **Day 2** inclusive.

National Stock Number: — Definition: National stock number of the component ISAAC uses this field to uniquely identify an item. The NSN field can contain any code as long as each unique item has a unique code. Identical SRUs common to two LRUs must have the same NSN value. Examples of NSN codes are PLSN, record number, a US stock number, or another countries stock number.Field: 18 character spaces. Input location: Kit Component Data Screen [cross reference with Nsn]

Nbases — Definition: The number of uniform bases the aircraft is stationed at. The model treats each base as though it is identical with every other base in terms of number of aircraft, flying hours and all other operational and maintenance factors. Field: 3 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Number of Bases:**]

Ndays — Definition: Number of days warning prior to the start of the war. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Number of Warning Days:**]

Negflag — Definition: This field name corresponds to the **Use Pre-specified Buy Quantity?** field from the Stock Options section of the Pipeline, Resupply, & Other Options Screen. Field: drop-down list box consisting of **Yes - Buy Quantity = Neglv:** and **No - Model Determines Quantity:**, 1 character space. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Use Pre-specified Buy Quantity?**]

- If **Use Pre-specified Buy Quantity?** [in the Pipeline, Resupply, & Other Options Screen] is set to **Yes - Buy Quantity = Neglv:**

  - the model will buy up from the respective ITASSE to NEGLV quantities sacrosanct for those components with a NOP value of either 'FIX' or 'NOP.'

  - the model will treat the respective NEGLV quantity as the floor for those components with a NOP value of 'AAA.'

  - the model will treat the respective NEGLV quantity as the quantity on order for those components with a NOP value of 'ORD.'

- If **Use Pre-specified Buy Quantity?** [in the Pipeline, Resupply, & Other Options Screen] is set to **No - Model Determines Quantity:**

  - the model will ignore the NEGLV quantity for those components with a NOP value of 'AAA' or 'ORD.'

  - the model will treat the respective NEGLV quantity as the ceiling for those components with a NOP value of 'NOP.'

- ► the model will buy up from the respective ITASSEto NEGLV quantities sacrosanct for those components with a NOP value ofFIX.'

Neglv — Definition: Negotiated level for this NSN or quantity of the component already on order as determined by the NOP field value. Field: 6 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with**Negotiated Level:**]

Negotiated Level: — Definition: Negotiated target level for this NSN or quantity of the component already on order as determined by the NOP field value.Field: 6 numeric spaces. Input location: Kit Component Data Screen [cross reference with Neglv]

– If **Use Pre-specified Buy Quantity?** [in the Pipeline, Resupply, & Options Screen] is set to **Yes - Buy Quantity = Neglv:**

- ► the model buys up from the respective ITASSE to NEGLV quantities sacrosanct for those components with a NOP value of either FIX' or 'NOP.'

- ► the model treats the respective NEGLV quantity as the floor for those components with a NOP value of 'AAA.'

- ► the model automatically buys the respective NEGLV quantity as the quantity on order for those components with a NOP value of 'ORD.'

– If **Use Pre-specified Buy Quantity?** [in the Pipeline, Resupply, & Options Screen] is set to **No - Model Determines Quantity:**

- ► the model ignores the NEGLV quantity for those components with a NOP value of 'AAA' or 'ORD.'

- ► the model treats the respective NEGLV quantity as the ceiling for those components with a NOP value of 'NOP.'

- ► the model buys up from the respective ITASSE to NEGLV quantities sacrosanct (i.e., automatically) for those components with a NOP value of 'FIX.'

NewValue* — Definition: The purpose of the **NewValue*** field is to allow the user to view the calculated new field value and to compare it to the current field value before applying it. The **NewValue*** is displayed to the right of the current field value. The current field value will have the field name in the field header. To the left of the current field value is the**Nsn** field. These records are sorted in NSN order. The **Nsn** field column is a reference point to show you which record the new and current field values correspond to. Input location: Preview/Edit/Apply page of Sensitivity Change page frame.

Next Higher Assembly: — Definition: Next higher assembly national stock number; the next higher assembly for LRUs will be the weapon system. Field: 18 character spaces. Input location: Kit Component Data Screen [cross reference with Nhansn]

Nhansn — Definition: Next higher assembly national stock number; the next higher assembly for LRUs will be the weapon system. Field: 18 character spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Next Higher Assembly:**]

Nmcs1 — Definition: This is the Not Mission Capable for Supply target or number of aircraft on ground (AOG) specified by the user for the first analysis day. The model purchases enough spares to meet that target or — if constrained by the budget — purchases the spares that yield the best spares mix without exceeding the budget constraint. Field: 5 character spaces; e.g., **4.0**). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **1st NMCS Target:**]

Nmcs2 — Definition: This is the Not Mission Capable for Supply target or number of aircraft on ground (AOG) specified by the user for the second analysis day. The model purchases enough spares to meet that target or — if constrained by the budget — purchases the spares that yield the best spares mix without exceeding the budget constraint. Field: 5 character spaces; e.g., **4.0**). Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with **2nd NMCS Target:**]

Nmcstarget — Definition: The NMCS target from the Model Parameters Screen. All days up through the **1st Analysis day** use the **1st NMCS Target**, otherwise the **2nd NMCS Target** is used. Field: 11 numeric spaces. Output location: Multi-day Evaluation Window [cross reference with **1st NMCS Target** and **2nd NMCS Target**]

No - Evaluate Selected Spares Mix Only — see Permit More Buys?

Non-Wartime **Flying Hrs/Sortie:** — Definition: Non-wartime flying hours per sortie. Field: text box. Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with Phr_sortie]

Non-Wartime — Consists of the **Total Flying Hours:** and **Flying Hrs/Sortie:** fields. Input locations: Kit Flying Hour Scenario Screen, Model Flying Hour Scenario Screen

Non-Wartime **Total Flying Hours:** — Definition: The non-wartime flying hour program for all weapon systems of interest. This field value will be divided by the **Number of Bases** and **Number** (of Weapon Systems) fields to determine the number of flying hours per aircraft. The **Total Flying Hours:** encompasses the non-wartime flying hour program only. Field: text box, 7 numeric spaces (2 decimal places). Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with Scen_day00→ Scen_day60 and Day]

NOP Flag: — Definition: Enables the user to specify non-optimal spares levels or to specify quantities of components that are already on order. Processing of this field is controlled by the **Use Pre-specified Buy Quantity** field of the Pipeline, Resupply, & Options Screen in conjunction with the NEGLV quantity from the component data record. Field: 3 character spaces (e.g., **NOP**, **FIX, ORD** or **AAA**). Input location: Kit Component Data Screen [cross reference with Nopflag]

- A value of 'NOP' will cause the model to buy exactly the quantity of the component specified in Neglv when the **Use Pre-specified Buy Quantity** is set to **Yes**. When the **Use Pre-specified Buy Quantity** is set to **No**, the model will treat the quantity specified in NEGLV as the ceiling for the respective component. In any case, components with an entry of 'NOP' will be included in the total budget but will not be included in the availability calculation.

- A value of 'FIX' will cause the model to buy exactly the quantity specified in NEGLV and will include the component in both the budget and availability calculations regardless of the **Use Pre-specified Buy Quantity** field value.

- A value of 'ORD' will cause the model to buy at least the quantity specified in NEGLV as the quantity of the respective component that is already on order and will be delivered prior to the analysis day when the **Use Pre-specified Buy Quantity** field value is **Yes**. When the **Use Pre-specified Buy Quantity** field value is **No** the model will ignore the component quantity specified in the respective NEGLV field.

- Any other value (e.g., 'AAA') will cause the model to treat the quantity specified in NEGLV as the floor for the respective component when the **Use Pre-specified Buy Quantity** is set to **Yes**. When the **Use Pre-specified Buy Quantity** is set to **No** the model will ignore the component quantity specified in the respective NEGLV field.

Nopflag — Definition: Enables the user to specify non-optimal spares options such as specifying a minimum spares target or buy total. Processing of this field is controlled by the **Use Pre-specified Buy Quantity** field of the Pipeline, Resupply, & Options Screen in conjunction with the NEGLV quantity from the Kit Component Data Screen. Field: 3 character spaces (e.g., **NOP**, **FIX, ORD** or **AAA**). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with NOP Flag]

- A value of 'NOP' will cause the model to buy exactly the quantity of the component specified in NEGLV when the **Use Pre-specified Buy Quantity** is set to **Yes**. In this case, components with an entry of 'NOP' will be included in the total budget but will not be included in the availability calculation. When the **Use Pre-specified Buy Quantity** is set to **No**, the model will treat the quantity specified in NEGLV as the ceiling for the respective component.

- A value of 'FIX' will cause the model to buy exactly the quantity specified in NEGLV and will include the component in both the budget and availability calculations regardless of the **Use Pre-specified Buy Quantity** field value.

- A value of 'ORD' will cause the model to treat the quantity specified in NEGLV as the quantity of the respective component that is already on order and will be delivered prior to the 2nd day of analysis whenthe **Use Pre-specified Buy Quantity** field value is **Yes**. When the **Use Pre-specified Buy Quantity** field value is **No** the model will ignore the component quantity specified in the respective NEGLV field.

- Any other value (e.g., 'AAA') will cause the model to treat the quantity specified in NEGLV as the floor for the respective component whenthe **Use Pre-specified Buy Quantity** is set to **Yes**. When the **Use Pre-specified Buy Quantity** is set to **No** the model will ignore the component quantity specified in the respective NEGLV field.

Not Reparable at Station — Definition: Percentage of base not repairable this station (NRTS) demands that are either condemned or sent to the depot for repair (overhaul) for this component. Field: Text Box with 8 numericspaces (5 decimal places). Input location: Kit Component Data Screen [cross reference with Bnrtsp and Bnrtsw]

Nsn — Definition: National stock number of the component ISAAC uses this field to uniquely identify an item. The NSN field can contain any code as long as each unique item has a unique code. Identical SRUs common to two LRUs must have the same NSN value. Examples of NSN codes are PLSN, record number, a US stock number, or another countries stock number. Field: 18 character spaces. Input location: Component Data Window; Output locations: Critical Item, Pipeline Data, Shopping List Data, Shop Comparison, and View Input-Output Windows. Location to edit information: Kit Component Data Screen. [cross reference with National Stock Number]

Nsn — Definition: The NSN is the field on the left side of the Preview/Edit/Apply page. The records on this page are sorted in NSN order. The **Nsn** field column is a reference point to show you which record the new and current field values correspond to. Input location: Preview/Edit/Apply page of Sensitivity Change page frame.

Number of Bases: — Definition: The number of uniform bases the aircraft is stationed at. The model treats each base as though it is identical with every other base in terms of number of aircraft, flying hours and all other operational and maintenance factors. Field: Text Box with 3 character spaces. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Nbases]

Number of Warning Days: — Definition: The number of days of warning before the start of the scenario (normally set to 0). In a model run that includes non-wartime (Day 0), this field indicates how many days the model will use wartime re-supply values before the start of the wartime scenario actually begins. Field: numeric. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Ndays]

Nunits — Definition: The number of weapon systems per base. This value is from the "Aircraft Number" field. The model assumes all bases have the same number of aircraft. Field: 3 character spaces. Input location: Run Log Window. Locations to edit information: Kit Parameters Screen or Model Parameters Screen. [cross reference with Aircraft Number]

Optimization: — Definition: The optimization parameter specifies the type of objective function used by the model. Field: Drop-down list box consists of **Confidence**, **ENMCS**, or **EBO/ENMCS** Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen.

- Confidence — Confidence level optimization works by maximizing the probability that the number of planes NMCS will not exceed a given target. This target is called the direct support objective (DSO). This probability is often called the confidence of meeting the DSO; marginal analysis to maximize this confidence is called "confidence optimization."

- ENMCS — The ENMCS optimization works by considering the confidence for all possible NMCS (from zero on up) and optimizing a weighted sum of them. Unfortunately, ENMCS is optimized only for the point on the curve where ENMCS equals the target. Thus, if you enter an NMCS target and a budget constraint and the resulting ENMCS is significantly different from the NMCS target, you need to rerun the model with a new NMCS target equal to the ENMCS result. Nevertheless, ISAAC produces the least expensive (optimal) spares mix needed to reach the ENMCS target. By experimenting with the target, ISAAC can also compute the spares mix that gives the lowest possible ENMCS for a specified cost. While maximizing confidence is not identical to minimizing ENMCS, the spares mix and performance results are not strikingly different. A spares mix built using confidence optimization normally yields a slightly better (higher) confidence level output but also a slightly worse (higher) ENMCS output and a worse (higher) EBO output.

- EBO/ENMCS — This optimization assumes cannibalization and uses two measures of performance: the primary measure — ENMCS — and a secondary one — EBOs. Though ENMCS optimization with cannibalization produces the maximum availability, it significantly increases maintenance workload, since the shop workers no longer get all spares from existing inventory (they must remove the cannibalizable parts from the failed assemblies). The spares that are cannibalized cause additional backorders and require the additional steps (and risk) involved in borrowing items from AOG aircraft and then reinstalling them later when the backorder is filled. Thus, the EBOs (included in the optimization with ENMCS) can be considered proportional to maintenance workload. The model then develops a compromise spares mix that produces slightly worse ENMCS output but significantly better EBO output (i.e., reduced maintenance workload) when compared to a pure ENMCS optimization.

Optmthd — Definition: Optimization method for this model run. The optimization parameter specifies the type of objective function used by the model.

Field: Drop-down list box consists of **ENMCS**, **Confidence**, or **EBO/ENMCS**, 1 character space. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with Optimization]

Order & Ship Time — Definition: Order and ship time in days for this component. This is the number of days from when a request is made on the depot for an item until that item is received in base supply. This does not include depot shortage time when a serviceable item is not available at the depot Field: 6 numeric spaces. Input location: Kit Component Data Screen [cross reference with Iostp and Iostw]

Os_start — Definition: The day that forward transportation from the depot starts. This is the value from the "Day Order and Ship Begins" field of the Pipeline, Resupply, & Options Screen. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with Day Order and Ship Begins]

Ospipe_1 — Definition: The order and ship pipeline on the first analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Ospipe_2 — Definition: The order and ship pipeline on the second analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Ospipe — Definition: The order and ship pipeline. Field: 7 numeric spaces (2 decimal places). Output location: Pipeline Data Window

Ost — Order and Ship Time (see Iostp & Iostw)

Other Options — Consists of the **Exponential Repair**; **Variance to Mean Ratio**; **Number of Bases**; and **Optimization** fields.

Pa_modelid — Definition: Unique number assigned by ISAAC to each model run for tracking and identification purposes in the library of previous runs. Field: 6 character spaces. Input location: Run Log Window [cross reference with Run #]

Pbuyl1 — Definition: The percentage of the LRU pipeline (either peak or on the first analysis day as specified by the **Purchase peak pipelines or max thru a given day** value**)** that the model buys sacrosanct to meet the requirement for the first analysis day. A value of 100 buys the whole pipeline, 50 buys half the pipeline, etc. An entry of 'ITEM' instructs the model to use the item-specific percentage as coded on the component data record. An entry of 'QPA' buys the pipeline for all items with a QPA greater than two. Field: 4 character spaces, **ITEM**, or **QPA**; Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **LRU % on 1st day:**]

Pbuyl2 — Definition: The percentage of the LRU pipeline (either peak or on the second analysis day as specified by the **Purchase peak pipelines or max thru a**

given day value**)** that the model buys sacrosanct to meet the requirement for the second analysis day. Field: 4 character spaces, **ITEM**, or **QPA**; Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **LRU % on second day:**]

Pbuys1 — Definition: The percentage of the SRU pipeline (either peak or on the first analysis day as specified by the **Purchase peak pipelines or max thru a given day** value**)** that the model buys sacrosanct to meet the requirement of the first analysis day. A value of 100 buys the whole pipeline, 50 buys half the pipeline, etc. An entry of 'ITEM' instructs the model to use the item-specific percentage as coded on the component data record. Field: 4 character spaces, **ITEM**, or **QPA**; Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **SRU % on 1st day:**]

Pbuys2 — Definition: The percentage of the SRU pipeline (either peak or on the second analysis day as specified by the **Purchase peak pipelines or max thru a given day** value**)** that the model buys sacrosanct to meet the requirement of the second analysis day. Field: 4 character spaces, **ITEM**, or **QPA**. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **SRU % on second day:**]

Permit More Buys? — Definition: Input location: Evaluation Setup Screen. [cross reference with Rb_runtype]

- **Yes - Use Mix as Starting Point (Add Spares if Needed to Reach Target)**. Definition: The model will add to the selected spares mix, if needed, to achieve the availability or cost targets specified on the Parameters Screen.

- **No - Evaluate Selected Spares Mix Only**. Definition: The model uses only the selected spares mix specified — no more or less spares. Since the evaluation only evaluates a single spares mix, no curve table is produced. The advantage of this option over the alternative is that with this, ISAAC can evaluate extreme cases that contain very unbalanced mixes of spares: too few or too many spares for some items.

    - **Standard 1 or 2 Day Run**. Definition: This is the standard evaluation run setting.

    - **Multi-day - All Days Inclusive (0,1,2,3,4...).** Definition: This will result in a multi-day evaluation run of the model. The model will evaluate the availability that this kit yields on each consecutive day to be analyzed. The model will evaluate the kit from **Day 1** to **Day 2** inclusive.

Phrsortie — Definition: This is the number of flying hours per non-wartime sortie. Field: 6 numeric spaces (3 decimal places). Input location: Run Log Window. Location to edit information: Flying Hour Scenario Screen. [Cross reference with **Flying Hrs/Sortie:** [non-wartime]]

Pltt — Definition: Procurement leadtime total for the item in months. The PLTT is the time from when an item is condemned to when a serviceable replacement for the item is procured and available at the base. The PLTT can be thought of as the sum of the administrative leadtime required to order the item once the failure is discovered, the production leadtime and the time required to process and ship the item. Location to edit information: Kit Component Data Screen. [cross reference with Procurement Lead Time]

- Field: 6 numeric spaces (1 decimal place). Input location: Component Data Window;

- Field: 6 numeric spaces. Output location: Shopping List Data Window;

Prev Runs button — Definition: Selecting this button opens the Find by Selected Field Window. This window contains the library of all previous ISAAC model runs (that have been run on your PC). Input location: Model Parameters Screen [Alternatively, you can use the two unlabeled buttons on either side of the **Prev Runs** button to move around the library of previous model runs without viewing the runs through the Find by Selected Field Window.] The buttons are defined as follows:

- Selecting the **>** button will fill the Model Parameters Screen with the run time parameters from the previous model run that was immediately above the currently displayed model run in the Run Library.

- Selecting the **>>** button will fill the Model Parameters Screen with the run time parameters from the run at the top of the Run Library.

- Selecting the **<** button will fill the Model Parameters Screen with the run time parameters from the previous model run that was immediately below the currently displayed model run in the Run Library.

- Selecting the **<<** button will fill the Model Parameters Screen with the run time parameters from the run at the bottom of the Run Library.

Preview / Edit / Apply Page — Definition: This page enables the user to: (1) preview the changes that were made on the Define Changes page; (2) edit the changes one record at a time; and (3) to apply the changes to the database. This page is accessed automatically when you select the **Calculate** button on the Define Changes page. Input location: Sensitivity Change page frame.

Previous Run Buy_Total Field from Shopping List — see Select Spares Mix From...

Print button —

- Definition: Selecting this button will initiatethe printing process of all the parameters on the Parameters Screen, Flying Hour Scenario Screen, and the Pipeline, Resupply, & Options Screen. Input location: Model Parameters Screen

– Definition: This button is not currently working on the Kit Parameter and Kit Component Data Screens. Input locations: Kit Component Data Screen, Kit Parameters Screen

Procure. Leadtime: — Definition: Procurement leadtime total for the item in months. The PLTT is the time from when an item is condemned to when a serviceable replacement for the item is procured and available at the base. The PLTT can be thought of as the sum of the administrative leadtime required to order the item once the failure is discovered, the production leadtime and the time required to process and ship the item. Field: 6 numeric spaces (1 decimal place). Input location: Kit Component Data Screen [cross reference with Pltt]

Purchase peak pipelines (T/F) or max thru a given day: — Definition: Specifies what pipeline values are to be bought sacrosanct to the level specified by LRU Percentage (on first and second days) and SRU Percentage (on first and second days). A value of 'F' indicates that the pipelines on the last day of the scenario will be used. A value of 'T' instructs the model to use the largest 'peak' pipeline over the scenario for each item. When "T" is selected the model computes the pipeline through each day to be analyzed for each item. The model purchases the pipeline for each item sacrosanct prior to conducting the marginal analysis for the kit. (The day on which each item attains its peak pipeline differs on the basis of the respective item's resupply/repair parameters.) Optionally, a specific day may be entered, as an integer from 0 to 99, forcing the model to use the maximum pipeline values through that day. Field:**T** (True) or **F** (False) or an integer. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Buypeak]

Qpanha — Definition: Quantity of the component installed on each unit of the respective next higher assembly. Field: 4 numeric spaces. Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with Quantity on Parent]

Qpchain — Definition: The total quantity of the component installed on the respective next higher assembly. This is the total quantity installed on this next higher assembly per aircraft. Not just the quantity on one of the next higher assemblies. Field: 4 numeric spaces. Output location: Critical Item Window.

Quantity on Parent: — Definition: Quantity of the component installed on each unit of the respective next higher assembly. Field: 4 numeric spaces. Input location: Kit Component Data Screen [cross reference with Qpanha]

Rb_buytype — Definition: This is the field where the selection from the Use Spares Mix as... section of the Evaluation Setup Screen is stored. There are two possible values which are listed below. Field: 1 numeric space. Input location: Run Log Window. Location to edit information: Evaluation Setup Screen. [cross reference with Use Spares Mix as...]

– 1 = **Forced Buys — Cost Included in Total Spares Cost** — Definition: the model includes the spares mix in the total spares cost (a budget estimate).

The model will purchase the same spares quantity (at the same price) as in the selected model run for each NSN that is contained in both runs. The model will use the previous run spares quantity as a floor.

– 2 = **Initial Assets — Spares are Free and NOT Included in Total Spares Cost —** Definition: the model does *not* include the spares mix in the total spares cost (a budget estimate). In other words, the model treats the spares mix as initial assets that have been previously procured and are not included in the total cost.

Rb_output — Definition: This is the field where the selection from the Select Spares Mix From... section of the Evaluation Setup Screen is stored. There are three possible values which are listed below. Field: 1 numeric space. Input location: Run Log Window. Location to edit information: Evaluation Setup Screen. [cross reference with Select Spares Mix From...]

– 1 = **Currently Selected Run — Buy_Total Field from Shopping List —** Definition: the model uses the spares mix, specifically the Buy Total value for each item, from the kit you selected at the very start of the evaluation process, from the previous run list.

– 2 = **Currently Selected Run — Recommended Buy (RMSSBUY) Field from Kit —** Definition: the model uses a previously stored spares mix solution as the starting point for the evaluation. One of the fields in the kit is called RMSSBUY (the recommended manufacturer system stock buy quantity). With this evaluation setting, the user can estimate the availability and cost of purchasing what the manufacturer recommends. Though the Rmssbuy field is set up for a standard initial procurement, where this information is assumed to be imported with the rest of the component level information, the user can use this field to specify any spares mix.

– 3 = **Previous Run — Buy_Total Field from Shopping List —** Definition: the model uses a spares mix, specifically the Buy Total value for each item, from an alternate run and possibly an alternate kit. Clicking this radio button opens the Find Model Run window. Select a model run from this window and the model will use the Buy_Total field value as the spares quantity to evaluate (for each NSN that the currently selected run and the previous run have in common). Since this option allows you to pick any run, you have to make sure there are some NSN numbers in common or else the model will cancel the operation. Also, if you originally selected a baseline kit to start the run, the model forces you to select a previous run since a baseline kit (i.e., a kit that has no associated model runs) contains no spares mix.

Rb_runtype — Definition: This is one of the two fields (the other field is Multiday) where the selection from the Permit More Buys? section of the Evaluation Setup Screen is stored. There are two possible values which are listed below. Field: 1 numeric space. Input location: Run Log Window. Location to edit information: Evaluation Setup Screen. [cross reference with Permit More Buys?]

– 1 = **Yes — Use Mix as Starting Point (Add Spares if Needed to Reach Target)** Definition: The model will add to the selected spares mix, if needed, to achieve the availability or cost targets specified on the Parameters Screen.

– 2 = **No - Evaluate Selected Spares Mix Only**. Definition: The model uses only the selected spares mix specified — no more or less spares. Since the evaluation only evaluates a single spares mix, no curve table is produced. The advantage of this option over the alternative is that with this, ISAAC can evaluate extreme cases that contain very unbalanced mixes of spares: too few or too many spares for some items.

Rdate — Definition: Run date, the date of the model run. Field: 8 date spaces (e.g., April 15, 1998 expressed as 04/15/98). Input location: Run Log Window

Recommended Buy: **—** Definition: The recommended manufacturer system stock buy quantity (RMSSBUY). With a standard initial procurement, this field value is what the manufacturer has recommended as a spares buy quantity. Input location: Kit Component Data Screen [cross reference with Rmssbuy]

Reset button — Definition: Input location: Define Changes page of Sensitivity Change page frame.

Resupply — Consists of the **Day Base Repair Begins** for: **RR LRU's**, **RRR LRU's** and **SRU's**; **Day Depot Repair Begins** for: **RR LRU's**, **RRR LRU's** and **SRU's**; **Day Order and Ship Begins**; and **Number of Warning Days** fields. Input locations: Kit Pipeline, Resupply, & Options Screen, Model Pipeline, Resupply, & Options Screen

Rmssbuy — Definition: The recommended manufacturer system stock buy quantity. With a standard initial procurement, this field value is what the manufacturer has recommended as a spares buy quantity. Field: 3 numeric spaces. Input location: Component Data Window. Location to edit information: Kit Component Data Screen. [cross reference with Recommended Buy]

Rtype — Definition: The type of reparable this spare is classified as. Field: Drop-down list box consists of **LRU** or **SRU**, 4 character spaces. Input location: Component Data Window; Output locations: Critical Item, Pipeline Data, Shopping List Data, and View Input-Output Windows. Location to edit information: Kit Component Data Screen.

Run Description — Definition: User entered descriptive statement used to identify the particular model run. This must be a unique description of the run (no two runs may have the same description). This information is entered in the Model Parameter Screen and displayed on the respective performance report. Field: character. Input location: Model Parameters Screen; Output location: Performance Report Window [cross reference with Describe]

Run Requirements button — Definition: Selecting this button starts the model processing to determine the requirements in terms of dollars and spares necessary to meet the optimization target expressed as ENMCS or confidence level or budget constraint. Input location: Model Parameters Screen

Run Date — Definition: Run date, the date of the model run. Field: 8 date spaces (e.g., April 15, 1998 expressed as 04/15/98). Output location: Performance Report Window; [cross reference with Run Date]

Run Evaluation button — Definition: Selecting this button opens the Evaluation Setup Screen. Input location: Model Parameters Screen

Run #— Definition: The run # is a unique number assigned to each model run for tracking and identification purposes in the library of previous runs. On the Model Parameters Screen this field will be **0** when you bring in a new baseline kit. If you click the **Modify** button this field will contain the model run number of the run you are modifying. Otherwise this field will contain the model run number corresponding to the model run currently open. Field: numeric spaces. Input location: Model Parameters Screen, Output location: Performance Report Window. [cross reference with Run #]

Run Date — Definition: Run date, the date of the model run. Field: numeric date (e.g., April 15, 1998 expressed as 04/15/98). Input Location: Model Parameters Screen. Output location: Performance Report Window [cross reference with Rdate and Run Date]

Run — Definition: This is a combination of the Run # and Run Description fields copied from the Model Parameters Screen. Field: character spaces. Input location: Evaluation Setup Screen

Runtype — Definition: The type of model run, either a requirements run or an evaluation model run. Field: **1** (Requirements) or **2** (Evaluation), 1 character space. Input location: Run Log Window. The model will enter this data as you choose **Run Requirements** or **Run Evaluation**.

Save button —

- Definition: Input location: Kit Parameters Screen
- Definition: Selecting this button will save the component data field information. If you don't select **Save** and move your cursor to another record you will lose the information you have entered since your last save. Input location: Kit Component Data Screen

Save Kit & Exit button — Definition: Input location: Define Changes page of Sensitivity Change page frame.

Scen_day00 ® Scen_day60 — Definition: The cumulative number of flying hours for all aircraft per respective day [e.g., Scen_day00 represents steady state (e.g., non-war conditions) ; Scen_day01 is the 1st day of the war; ...] that the fleet under consideration will be operational. The flying hour program per day for all weapon systems of interest. This field value will be divided by the **Number of Bases** and **Number** (of Weapon Systems) fields to determine the number of flying hours per day per aircraft. The scenario encompasses both the non-wartime and wartime flying hour programs. Only one value may be entered for the non-wartime flying hour program. Multiple values may be entered for the wartime flying hour program — however, only one value may be entered for each day. Field: 7 numeric spaces (2 decimal places). Input location: Run Log Window; Location to edit information: Flying Hour Scenario Screen. [cross reference with **Total Flying Hours:** (for non-wartime) and **Day 01 – 60** (for wartime)]

Scenario Baseline button **—** Definition: Selecting this button opens the Kit Flying Hour Scenario Screen. Input location: Kit Parameters Screen

Scenario button — Definition: Selecting this button opens the Model Flying Hour Scenario Screen. Input location: Model Parameters Screen

Schmaint — Definition: Scheduled demand per flying hour for this component. The programmed maintenance generated demand per flying hour. Field:8 numeric spaces (5 decimal places). Input location: Component Data Window. Currently, this field can not be viewed or edited within ISAAC.

Select Spares Mix From... — Definition: This group of options provides the user with flexibility in selecting a spares mix to evaluate. This top section of the Evaluation Setup Screen consists of the following three radio buttons that allow the user to specify the model run and particular field that the spares quantity will come from. Input location: Evaluation Setup Screen. [cross reference with Rb_output]

- **Currently Selected Run - Buy_Total Field from Shopping List** — Definition: the model uses the spares mix, specifically the Buy Total value for each item, from the kit you selected at the very start of the evaluation process, from the previous run list.

- **Currently Selected Run - Recommended Buy (RMSSBUY) Field from Kit** — Definition: the model uses a previously stored spares mix solution as the starting point for the evaluation. One of the fields in the kit is called RMSSBUY (the recommended manufacturer system stock buy quantity). With this evaluation setting, the user can estimate the availability and cost of purchasing what the manufacturer recommends. Though the Rmssbuy field is set up for a standard initial procurement, where this information is assumed to be imported with the rest of the component level information, the user can use this field to specify any spares mix.

- **Previous Run Buy_Total Field from Shopping List** — Definition: the model uses a spares mix, specifically the Buy Total value for each item, from an alternate run and possibly an alternate kit. Clicking this radio button opens the Find Model Run Window. Select a model run from this window and the model will use the Buy_Total field value as the spares quantity to evaluate (for each NSN that the currently selected run and the previous run have in common). Since this option allows you to pick any run, you have to make sure there are some NSN numbers in common or else the model will cancel the operation. Also, if you originally selected a baseline kit to start the run, the model forces you to select a previous run since a baseline kit (i.e., a kit that has no associated model runs) contains no spares mix.

Sensitivity Change page frame— This is the vehicle for making global changes to the kit component database. The Sensitivity Change page frame contains three pages: the Define Changes page, the Preview / Edit / Apply page, and the View Kit page. As the page names imply: field data changes are defined on the Define Changes page, previewed and applied on the Preview / Edit / Apply page, and the kit component database can be viewed via the View Kit page.

Set Wartime Flying Hours for a Range of Days bar — Definition: Selecting this bar opens the Enter a Range of Days and the Value to Store dialog box. This dialog box is used to set the flying hours as a constant over a range of days in wartime. Input locations: Kit Flying Hour Scenario Screen, Model Flying Hour Scenario Screen

Shiftcond — Definition: The average number of condemnations that occur after the aircraft delivery year. The Shiftcond spares are spares not required in the first coverage year so the model can delay ordering them without impacting the availability estimates. The model delays approximately the same percentage of spares (via Shiftcond) across all items in an effort to match the annual budget constraint. (Note: the Shiftcond quantity is always less than or equal to the Buy_total quantity.) Field: 7 numeric spaces. Output location: Shopping List Data Window

Shiftpipe — Definition: Not used by the model. Field: 7 numeric spaces. Output location: Shopping List Data Window

Sortie_flown — Definition: This is the number of sorties that the fleet could sustain given the constraints of **Max. Sorties/Day:**, **Flying Hrs/Sortie:**, and the number of aircraft expected to be available on the specified day of the multi-day evaluation. It is calculated as (Flying hours flown / (Number of Aircraft - ENMCS) * (Hours / Sortie)). Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Sortie_plan — Definition: This is the planned number of sorties for the fleet on the particular day of the multi-day evaluation. It is calculated as (Flying hours planned / (Number of Aircraft * (Hours / Sortie)). Field: 11 numeric spaces (2 decimal places). Output location: Multi-day Evaluation Window.

Spares **Coverage Period:** — Definition: This is the time during which the initial provisioning system is responsible for procuring spares for this weapon system. ISAAC estimates the spares required for that entire time period. The sole impact of increasing the coverage period is to increase the forecasted number of condemnations we consider and thus the number of spares required to replace those condemnations. Using only the expected condemnations in an item's PLTT (the basic assumption used in replenishment models) underestimates the condemnations if the coverage period is longer than the PLTT and overestimates them if it is shorter. Thus, ISAAC uses an additional accounting factor called CondAssets where the sum of the CondAssets and the condemnation pipeline equals the expected number of condemnations in a coverage period. After the coverage period, the model assumes the standard replenishment system can provide spares support. Field: numeric. Input location: Kit Parameters Screen; Model Parameters Screen [cross reference with Buycover] The user must also consider three additional factors when entering a coverage period:

- The user must set the **Include Starting Assets?** drop-down list box to **Use Assets** in the advanced parameters (Chapter 3) to procure spares for the entire coverage period. If you set the **Include Starting Assets?** drop-down list box to **- Do Not Use Starting Assets**, the model will not allow you to enter a coverage period greater than 0 until the **Use Assets** option is selected.

- The user cannot enter a coverage period value between 0 and 1 year, because that time frame includes dynamic flying hours (see Figure 1-6) and thus is inconsistent with the basic assumption of steady-state flying hours that the model makes for the non-wartime period.

- If the user enters a “0”, the model treats that value not as a time period but as a switch to use a different logic in calculating condemnations. A zero causes ISAAC to ignore the coverage period and its adjustments and estimate condemnations solely on the basis of each item’s PLTT.

SRU % on second day: — Definition: The percentage of the SRU pipeline (either peak or on the second analysis day as specified by the **Purchase peak pipelines (T/F) or max thru a given day:** value**)** that the model buys sacrosanct to meet the requirement of the second analysis day. Field: numeric (1 decimal place) or **ITEM**. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Pbuys2]

SRU % on first day: — Definition: The percentage of the SRU pipeline (either peak or on the first analysis day as specified by the **Purchase peak pipelines (T/F) or max thru a given day:** value**)** that the model buys sacrosanct to meet the requirement of the first analysis day. A value of 1.0 buys the whole pipeline, 0.5 buys half the pipeline, etc. An entry of ‘ITEM’ instructs the model to use the item-specific percentage as coded on the component data record. Field: numeric (1 decimal place) or **ITEM**; Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen [cross reference with Pbuys1]

Stk — Definition: This field is only used during an evaluation model run (currently not operational). Field: 2 character spaces (e.g., numeric or *****). Input location: Run Log Window. This field can not be edited within ISAAC.

Stkopt — Definition: Specifies whether existing stock is considered in the computations. Field: **1** (Do Not USE Starting Stock) or **2** (Use Assets - Item's

Itasse), 1 numeric space. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with Include Starting Assets?]

Stock Options — Consists of the **Include Starting Assets?**; **Use Pre-specified Buy Quantity?**; **Force Buy Based on Pipeline % Below**: **LRU % on first day**, **LRU % on second day**, **SRU % on first day**, **SRU % on second day** ; and **Purchase peak pipelines (T/F) or max thru a given day** fields. Input locations: Kit Pipeline, Resupply, & Options Screen, Model Pipeline, Resupply, & Options Screen

Stock / Resupply / Other Options button — Definition: This is the button used to open the Stock, Resupply, & Options Screen of the model and the kit. Input locations: Kit Parameters Screen; Model Parameters Screen

Targ1— Definition: This is the number of spares purchased by component minus those spares that are expected to be condemned during the coverage period for model run #1 of the comparison. Field: 7 numeric spaces. Output location: Shop Comparison Window.

Targ2— Definition: This is the number of spares purchased by component minus those spares that are expected to be condemned during the coverage period for model run #2 of the comparison. Field: 7 numeric spaces. Output location: Shop Comparison Window.

TargDelta— Definition: The difference in item target quantity between model runs 1 & 2. This is the absolute value of the difference between Targ1 and Targ2. Field: 4 numeric spaces. Output location: Shop Comparison Window.

TARGET — Definition: This is the number of spares purchased for all components minus those spares that are expected to be condemned during the coverage period. Consists of four distinct parameters: average, minimum, maximum and sum. Fields — **Average**, **Minimum**, and **Maximum:** 7 numeric spaces each; **Sum**: 10 numeric spaces (2 decimal places). Output location: Statistics window

Target — Definition: This is the number of spares purchased by component minus those spares that are expected to be condemned during the coverage period. Field: 7 numeric spaces. Output locations: Critical item, Shopping List Data, and View Input-Output Windows

Tasse — Existing Assets *(see Itasse)*

TOIMDRP — Definition: Non-war demand per flying hour for all components. Consists of four distinct parameters: average, minimum, maximum and sum. Fields —**Average**, **Minimum**, and **Maximum:** 8 numeric spaces (5 decimal places) each; **Sum**: 10 numeric spaces (5 decimal places). Output location: Statistics Window

Toimdrp — Definition: Non-wartime demand per flying hour for this component. The demand is based on the expected number of failures for this item. An item is classified as failing if it can only be recovered through procurement or repair. Field: 8 numeric spaces (5 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Failure Based Demand**]

Toimdrw — Definition: Wartime demand per flying hour for this component. The demand is based on the expected number of failures forthis item. An item is classified as failing if it can only be recovered through procurement or repair. Field: 8 numeric spaces (5 decimal places). Input location: Component Data Window; Location to edit information: Kit Component Data Screen. [cross reference with **Failure Based Demand**]

Total Combined (Day 1 and Day 2) Buy Cost —Definition: This equals the sum of the buy cost for each analysis day. Field: 11 numeric spaces. Output location: Performance Report Window

Total Flying Hours: — see Non-Wartime **Total Flying Hours:**

Total Combined (Day 1 and Day 2) Buy Cost — Definition: This equals the sum of the buy cost for each analysis day. Field: 11 numeric spaces (2 decimal places). Output location: Performance Report Window

TotDelta — Definition: The absolute value of the difference between BuyTot1 and ButTot2. This is the difference in total spares buy quantities (by item) between the two model runs under comparison. Field: 4 numeric spaces. Output location: Shop Comparison Window

Totlrus1 — Definition: The total number of LRUs considered for purchase in support of the first analysis day. If an LRU has an excess number of spares, the model may ignore it in the calculation to speed processing time and not include it in this table. Consists of one distinct field value for each budcode and a SUM across all budcodes. Field: 4 numeric spaces. Output location: Performance Report Window

Totlrus2 — Definition: The total number of LRUs considered for purchase in support of the second analysis day If an LRU has an excess number of spares, the model may ignore it in the calculation to speed processing time and not include it in this table. Consists of one distinct field value for each budcode and a SUM across all budcodes. Field: 4 numeric spaces. Output location: Performance Report Window

Totpipe_1 — Definition: This is the sum of the base pipeline mean (Baspipe_1 value) plus the depot pipeline mean (Deppipe_1 value) plus the order and ship pipeline mean (Ospipe_1) plus the condemnation pipeline mean (Conpipe_1) and the awaiting parts pipeline mean (AWP_1 for those components that have lower indenture items associated with them) from the Pipeline database for the first analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

Totpipe_2 — Definition: This is the sum of the base pipeline mean (Baspipe_2 value) plus the depot pipeline mean (Deppipe_2 value) plus the order and ship pipeline mean (Ospipe_2) plus the condemnation pipeline mean (Conpipe_2) and the awaiting parts pipeline mean (AWP_2 for those components that have lower indenture items associated with them) from the Pipeline database for the second analysis day. Field: 7 numeric spaces (2 decimal places). Output location: Critical Item Window

TOTPIPE — Definition: This is the sum of the base pipeline mean (BASPIPE) plus the depot pipeline mean (DEPPIPE) plus the condemnation pipeline mean (CONPIPE) and the awaiting parts pipeline mean (AWP for those components that have lower indenture items associated with them) from the Pipeline database for the day(s) to be analyzed. Consists of four distinct parameters: average, minimum, maximum and sum. Fields —**Average**, **Minimum**, and **Maximum:** 8 numeric spaces (5 decimal places) each; **Sum**: 10 numeric spaces (5 decimal places). Output location: Statistics Window

Totpipe — Definition: This is the sum of the base pipeline mean (Baspipe value) plus the depot pipeline mean (Deppipe value) plus the order and ship pipeline mean (Ospipe) plus the condemnation pipeline mean (Conpipe) and the awaiting parts pipeline mean (AWP for those components that have lower indenture items associated with them) from the Pipeline database for the day(s) to be analyzed. Field: numeric (2 decimal places). Output locations: Pipeline Data and View Input-Output Windows

Type: — Definition: The type of component this spare is classified as. All components that are not at the highest indenture level are classified as SRUs. Field: Drop-down list box consists of **LRU** or **SRU**. Input location: Kit Component Data Screen [cross reference with Type]

Undo button — Definition: Selecting this button undoes the previous parameter changes and brings back the original parameters (on the Parameters Screen, Pipeline, Resupply, & Options Screen, and the Flying Hour Scenario Screen) of the selected kit or model run. Input locations: Kit Parameters Screen, Model Parameters Screen

Use Pre-specified Buy Quantity? — Field: Drop-down list box consists of **Yes - Buy Quantity = Neglv** or **No - Model Determines Quantity**. Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen

– If **Use Pre-specified Buy Quantity?** is set to **Yes - Buy Quantity = Neglv:**

  ► the model will buy up from the respective ITASSEto NEGLV quantities sacrosanct (i.e., automatically) for those components with a NOP value of either 'FIX' or 'NOP.'

  ► the model will treat the respective NEGLV quantity as the floor for those components with a NOP value of 'AAA.'

- the model will treat the respective NEGLV quantity as the quantity on order for those components with a NOP value of 'ORD.'

– If **Use Pre-specified Buy Quantity?** is set to **No - Model Determines Quantity:**

- the model will ignore the NEGLV quantity for those components with a NOP value of 'AAA' or 'ORD.'

- the model will treat the respective NEGLV quantity as the ceiling for those components with a NOP value of 'NOP.'

- the model will buy up from the respective ITASSEto NEGLV quantities sacrosanct (i.e., automatically) for those components with a NOP value of 'FIX.'

Use Spare Mix as... — Definition: This group of options provides the user with flexibility in using the spares mix just selected. The user can either treat the spares mix as part of the total spares to be purchased or as "free" initial assets. This middle section of the Evaluation Setup Screen consists of the following radio button options. Input location: Evaluation Setup Screen. [cross reference with Rb_buytype]

– **Forced Buys - Cost Included in Total Spares Cost** — Definition: the model includes the spares mix in the total spares cost (a budget estimate). The model will purchase the same spares quantity (at the same price) as in the selected model run for each NSN that is contained in both runs. The model will use the previous run spares quantity as a floor.

– **Initial Assets - Spares are Free and NOT Included in Total Spares Cost** **—** Definition: the model does *not* include the spares mix in the total spares cost (a budget estimate). In other words, the model treats the spares mix as initial assets that have been previously procured and are not included in the total cost.

Usedecel — Definition: This is where the results of the **Decelerate Hrs .. selection are recorded. This field must be T (True) in order to use the decelerated flying hour capability of ISAAC to translate non-wartime demand into wartime demand. When this field is T, the Decelerate field is used to decelerate flying hour based demand to the wartime demand rate.** Field: 1 logical space, **T** or **F**. Input location: Run Log Window [cross reference with **Decelerate Hrs..**]

User: — Definition: User entered name information. Input location: Model Parameters Screen [cross reference with Asm_user]

Usnsn — Definition: United States national stock number of the component Field: 15 character spaces (actual NSNs occupy 13 or 15 spaces while dummy

NSNs occupy 15 spaces with zeroes). Input location: Component Data Window; This field can not be edited within ISAAC.

Value: — Definition: This is a text box where the user enters the fleet daily flying hour value that will be applied to each war day from the **First Day:** through the **Last Day:**. Field: 5 numeric spaces (2 decimal places). Input location: Enter a Range of Days and the Value to Store dialog box (accessed through the Flying Hour Scenario Screen).

Variance to Mean Ratio: — Definition: The variance-to-mean ratio (VMR) is a parameter used to adjust an item's demand uncertainty. The higher the value, the greater the uncertainty, and the greater the spares requirement needed to meet the target, all else being equal. From this field, the VMR is set for all component demands from this field. The default VMR of "1.00" indicates a random or Poisson process. Sometimes actual demands can be more erratic than those generated by a Poisson process or we may want to take into account the uncertainty about demand rates that is caused by inherent difficulty in forecasting the future. In either case, we should use a probability distribution that has more variance than Poisson; i.e., VMRs greater than 1.00. For those cases, the model uses a negative-binomial distribution with a variance larger than the mean. Currently, ISAAC will accept VMRs from 1.00 to 7.00 only (i.e., Poisson and negative-binomial probability distributions for component demand.) Input location: Kit Pipeline, Resupply, & Options Screen; Model Pipeline, Resupply, & Options Screen.

View (or View Only) — View is the default mode when you enter the Model Parameters Screen. While in the view mode you cannot edit the respective window or screen or any of their field values.

- In both the Kit and the Model portions of ISAAC: The View mode applies to the Model Parameters Screen, the Flying Hour Scenario Screen and the Pipeline, Resupply, and Options Screen.

- The View mode cannot be applied to the Component Data Screen of the Kit portion of ISAAC.

View Kit page — The View Kit page enables the user to view the kit database in real time. Once changes are applied on the Preview/Edit/Apply page they are input to the respective kit database. The View Kit page is a view only page that will not allow you to edit any of the displayed data. Input location: Sensitivity Change page frame.

Vmr — Definition: Variance to mean ratio of the component's demands. The variance to mean ratio (VMR) is a parameter used to adjust an item's demand uncertainty. The higher the value the greater the uncertainty and the greater spares requirement needed to meet the target all else being equal. The VMR is set for all component demands from this field. The default VMR of "1.00" is indicative of a random or Poisson process. Sometimes actual demands can be more erratic than a Poisson process or we may want to consider uncertainty about the demand rates caused by the inherent difficulty in forecasting the future. In either case, we should use a probability distribution that has more variance than Poisson or VMRs greater than one. For those cases, the model uses a negative-binomial distribution with a variance larger than the mean. Currently, ISAAC will only accept VMRs from 1.00 to 7.00 (i.e., Poisson and negative-binomial probability distributions for component demand.)Field: 4 character spaces. Input location: Component Data and Run Log Windows; Location to edit component level information: Kit Component Data Screen. [cross reference with **Variance to Mean Ratio:**]

Waitdays1 — Definition: Field: 7 numeric places (2 decimal places). Output location: Critical Item Window

Waitdays2 — Definition: Field: 7 numeric places (2 decimal places). Output location: Critical Item Window

Warn — Definition: The number of days warning before the start of the scenario (normally set to 0). In a model run that includes non-wartime (Day 0), this field indicates the number of days the model will use wartime re-supply values prior to the start of the wartime scenario. Field: 2 character spaces. Input location: Run Log Window. Location to edit information: Pipeline, Resupply, & Options Screen. [cross reference with **Number of Warning Days:**]

Wartime **Max Sorties/Day:** — Definition: The maximum number of sorties per plane per day in wartime. Field: text box, Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with Maxsorties]

Wartime Flying Hours [Day 01 – 60] — Definition: Field: 60 separate text boxes, one for each day. Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with Scen_day01→ Scen_day60 and Day]

Wartime **Flying Hrs/Sortie:** — Definition: Wartime flying hours per sortie. Field: text box., Input location: Kit Flying Hour Scenario Screen; Model Flying Hour Scenario Screen [cross reference with Whr_sortie]

Wartime — Consists of the **Max Sorties/Day:** and **Flying Hrs/Sortie:** fields. Input locations: Kit Flying Hour Scenario Screen, Model Flying Hour Scenario Screen

Weapon System: — Definition: This field is entered by the user when you define a kit (copy, new or import). The Weapon System is used by the model to identify the kit information. ISAAC inserts your Weapon System name into the Next Higher Assembly Field of each LRU component record. Field: character (e.g., F15I). Location to input information: Define Kit ID dialog box; Input locations: Kit Parameters Screen and Model Parameters Screen. Output location: Performance Report Window. [cross reference with Wsname]

Whrsortie — Definition: This is the number of flying hours per wartime sortie. Field: 6 numeric places (3 decimal places). Input location: Run Log Window. Location to edit information: Flying Hour Scenario Screen [cross reference with **Flying Hours / Sortie: [wartime]**]

Wpnsys — Definition: For items that are unique to this weapon system, this field will have the same value as the NHANSN for an LRU. For items that are common to more than one IAF weapon system (what we call MALHA items), enter 'OTHER' in this field. Field: 15 character spaces. Input location: Component Data Window; This field can not be edited within ISAAC. [cross reference with Weapon System]

Wsname — Definition: This field is entered by the user when you define a kit (copy, new or import). The field value is entered in the Weapon System field of the Define Kit ID dialog box. The Weapon System is used by the model to identify the kit information. ISAAC will insert your Weapon System name into the Next Higher Assembly Field of each LRU component record.For items that are unique to this weapon system, this field will have the same value as the NHANSN for an LRU. For items that are common to more than one IAF weapon system (what we call MALHA items), enter 'OTHER in this field. Field: 13 character spaces. Input location: Run Log Window; Output location: Performance Report Window. Location to edit this information is the Define Kit ID window [cross reference with Weapon System]

Yes - Use Mix as Starting Point (Add Spares if Needed to Reach Target) — see Permit More Buys?